FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Improving geolocation by combining GPS with Image Analysis

Fábio Filipe Costa Pinho

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Alexandre Miguel Barbosa Valle de Carvalho (PhD)

12^{th} July, 2013

Improving geolocation by combining GPS with Image Analysis

Fábio Filipe Costa Pinho

Mestrado Integrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

Chair: Doctor Rui Carlos Camacho de Sousa Ferreira da Silva
External Examiner: Doctor José Manuel de Castro Torres
Supervisor: Doctor Alexandre Miguel Barbosa Valle de Carvalho

---

12^{th} July, 2013

Abstract

In recent years, geolocation technologies have evolved so fast, that today almost every technological mobile device can be traceable through a simple internet connection. A few years ago, geolocation system was almost another term for Global Positioning System (GPS) receivers, because that was almost the only known technology of that kind. With the technological evolution over the years, new geolocation technologies emerged, and the existing ones became more accurate, accessible and portable. Today it is possible to find cheaper GPS Navigation Systems, and much more accurate and faster than a decade ago. Most smartphones already have an incorporated GPS receiver, sometimes combined with network data, and web browsers can already track (even though with low accuracy) the actual location of some IP address.

Despite all this evolution, geolocation by GPS can fail, due to lack of visibility to the satellites: without direct view from the receiver to a set of at least 4 satellites is not possible to correctly execute the necessary triangulation. To resolve this problem, GPS might be combined with network information (A-GPS), which allows a faster triangulation and an accurate response.

Given this context, this dissertation follows an innovative approach in geolocation that combines GPS information with a computer vision component. The main goal is to demonstrate that geolocation can sometimes be more accurate with the help of an image analysis system, which adds value to the coordinates read from the GPS by reducing the error through an identification of georeferencial entities in captured frames.

With this approach, it would be possible for a device to know its relative position, even in locations where the GPS information is not available, fixing some known problems like the positioning inside Urban Canyons, where the GPS information is unavailable for moments leaving geolocation softwares with no clue about their position.

To demonstrate the validity of this concept a prototype was implemented and used for a series of tests. This prototype consists in an application aimed for public transports, developed for Fraunhofer AICOS, institution focused in the improvement and usability of the information and communication technologies (ICT), mainly of the senior population, with application in mobility. The implemented prototype has the goal of improve the visibility of a public transport passenger to the exterior of the vehicle, using a mobile application that reproduces the exterior landscape, signaling a sequence of points of interest (POI) and adding textual information about that POI to the image.

With the help of computer vision, the lack of geolocation may be compensated and, as can be seen in the evaluation chapter, the system, if able to, know its location even when GPS information lacks or is outdated.

Resumo

As tecnologias de geolocalização têm evoluído tão rapidamente, que hoje em dia quase todos os dispositimos tecnológicos podem ser localizados através de uma simples ligação à internet. Há alguns anos atrás, os sistemas de localização geográfica resumiam-se praticamente aos vulgares recetores GPS, que eram a tecnologia de localização geográfica mais conhecida. Com os avanços da tecnologia, foram surgindo novos sistemas de geolocalização, e os já existentes tornaram-se cada vez mais precisos, acessíveis e portáteis. Hoje em dia, é possível encontrar sistemas de navegação GPS muito mais baratos, precisos e rápidos, que há uma década atrás. Muitos *smartphones* trazem já incorporado um recetor GPS, que pode funcionar inclusivamente combinado com informação proveniente da rede móvel, e os navegadores *web* (também conhecidos como *browsers*) podem também localizar, ainda que com má precisão, a posição atual de alguns endereços de IP.

Apesar de toda esta evolução, a geolocalização por GPS ainda falha, devido à falta de visibilidade dos satélites por parte dos recetores: sem vista direta do recetor GPS para um mínimo de quatro satélites não é possível executar uma correta triangulação da posição do recetor. Para resolver este problema, a informação de GPS é, por vezes, combinada com informação proveniente da rede, permitindo obter uma triangulação mais precisa e uma resposta substancialmente mais rápida.

Dado este contexto, esta dissertação pretende seguir uma abordagem inovadora na geolocalização, combinando informação de GPS com uma componente de visão por computador. O principal objetivo é demonstrar que a geolocalização pode, em determinadas situações melhorar, com a ajuda de um sistema de análise de imagem, que adicione valor às coordenadas lidas do recetor GPS, através da identificação de entidades nas imagens capturadas.

Com este sistema, seria possível um dispositivo saber a sua posição relativa, mesmo em localizações onde a informação de GPS não se encontre disponível, resolvendo alguns problemas característicos da geolocalização por GPS, nomeadamente o caso do posicionamento em "Urban Canyons". Nestas localizações, onde existem enormes arranha-céus que dificultam a receção de sinal, os sistemas de geolocalização ficam normalmente sem saber a sua atual posição, fator que leva a um natural mau funcionamento.

Para demonstrar a validade deste conceito, foi implementado um protótipo, posteriormente usado numa série de testes. Este protótipo consiste numa aplicação direcionada aos transportes públicos, desenvolvida para o Fraunhofer AICOS, instituição que se foca na melhoria e usabilidade das tecnologias de informação e comunicação (TIC), nomeadamente da população infoexcluída (principalmente os seniores), e com aplicação na mobilidade (individual ou coletiva). O protótipo implementado tem o intuito de melhorar a visibilidade para o exterior, dos passageiros de um transporte público, recorrendo ao uso de uma aplicação móvel que reproduz a paisagem exterior e assinala uma sequência de pontos de interesse, acrescentando à imagem informação textual sobre os mesmos.

Com a ajuda de um sistema de visão por computador, a falta de geolocalização por GPS pode ser compensada e, tal como pode ser visto no capítulo de avaliação, o sistema, caso funcione

devidamente, consegue saber a sua localização, mesmo quando há falta de informação GPS ou esta se encontra desatualizada.

Acknowledgements

First of all I would like to thank the Fraunhofer AICOS institute and all its professionals for all the support and confidence they had shown, providing me an unique work environment.

My special thank Professor Alexandre Carvalho, not only for all his advices and suggestions, which revealed very helpful during this dissertation, but also for all the patience and encouragement. Your help was essential, and much more than simple guidance.

Then, a special thank to my supervisor at Fraunhofer, Rui Carreira, for his guidance, patience and time availability.

An unique and very special thank to my best friend and girlfriend Joana Maia for all the support given throughout my life, and for all the believe and encouragement you gave me in the hardest moments. Thank you for inspire me everyday, and for always making me give the best of me.

Last but not least, I would like to thank to my family and friends for your comprehension and support. Your help was, with no doubts, very important, and without your support this would not be possible. Thank you for being there, and for accompany me during this journey.

Porto, 17^{th} June, 2013

Fábio Pinho

"Don't worry about failure,
you only have to be right once."

Drew Houston, Dropbox

Contents

List of Figures

List of Tables

Abbreviations

3G Third Generation

4G Fourth Generation

A-GPS Assisted Global Positioning System

CV Computer Vision

CV-GPS Computer Vision plus GPS

GPS Global Positioning System

HTTP HiperText Transfer Protocol

IA Image Analysis

ICT Information and Communication Technologies

IP Internet Protocol

Km Kilometers

LAN Local Area Network

MJPEG Motion JPEG

PoE Power over Ethernet

POI Point of Interest

PRN PseudoRandom Noise

RTSP Real Time Streaming Protocol

Chapter 1

Introduction

1.1 Motivation

Currently, geolocation technologies have an important role in many technological fields, from simple navigation systems, found today in a considerable number of transport vehicles, to applications in agriculture (more precise dispersion of chemicals), aviation (increasing safety and efficiency of flight), marine activities (like search and rescue, measuring speed and tracking the mariners location), and in many other fields including, for instance, Recreation or Public Safety and Disaster Relief, or even in the determination of the satellites orbit (space-qualified GPS units). [USA11]

In recent years, the means used in the calculation of the geographic localization have evolved and become progressively more accurate. The number of methods and technologies used for geolocation combining, for instance, data received from GPS (Global Positioning System) satellites with data from cellular networks, can give an accurate earth position, but not an exact position. An accurate positioning of a certain location corresponds, in this scenario, to the geographical coordinates of that location, with a small error that do not influence the geolocation itself. An acceptable error, may be close to 10 or 15 meters, in which case the geolocation may be considered accurate. Factors like noise or the lack of direct view from the receiver to the satellites, makes difficult to track the receivers position, moreover if the receiver is moving at a considerable velocity, in an urban environment. The well known Urban Canyons, manifested by streets cutting through dense blocks of structures, especially skyscrapers that form a human-built canyon, are a good example of a possible GPS system failure, because the reception of the radio signal is mostly affected.

The motivation for this work relies in the opportunity to explore computer vision capabilities to improve geolocation, especially under very dense urban environments (situations where GPS sensors loose accuracy) using this knowledge in the development of an application prototype capable of solving an existing problem in transport vehicles: the lack of good vision that a public transport passenger has, to the exterior landscape and the lack of personalized access to information about the points of interest in that landscape.

1.2 Problem Description

The problem consists in trying to solve the loss of precision in geolocation when GPS fails to provide accurate data, such as in specific environments like Urban Canyons. Factors like noise or the lack of direct view from the receiver to satellites, make difficult to keep the receivers position accurate, moreover if the receiver is moving at a considerable velocity, in an urban environment. This problem leads to small deviations in the position calculated by the receivers, resulting in temporary loss of precision. In a commercial GPS, this problem is frequently compensated by the software, which predicts the current vehicle position, but sometimes the software can not fix the position, especially if the users behaviour is unexpected (such as exiting the highway to a parallel road).

The accuracy of the GPS is more critical in applications that need to know if the receiver already passed by some specific position. In that cases, a high GPS error can position the receiver already in front of some location, when it may be yet dozens of meters behind. That simple case might be solved with a simple computer vision component complementing GPS information, so that in case of GPS failure, the geolocation might be obtained from the performed image analysis.

1.3 Objectives

Given the problem presented in the previous section, the main goal of this work is the search for a solution that is able to improve geolocation by using an image analysis module capable of compensate the GPS receivers failure. This way, it is necessary to demonstrate that geographical location accuracy can be improved with the help of simple computer vision algorithms. To do so, a prototype will be developed, demonstrating the use of the system above explained in the resolution of a real problem.

By combining GPS information with an image analysis system, it might be possible to obtain more conclusive answers about the proximity to a well known position, as well as detect if the system already passed by that position or not. Certain buildings, roads or monuments are characteristic in a city, and may be easily recognized by image descriptors. Those characteristic points may be called Points of Interest (POI). Every point of interest in earth surface can have a specific GPS coordinate associated, which means it is possible to assume, that if some POI (for instance a building or road) is detected in an image, the system is actually close to the GPS coordinates of that POI.

To demonstrate that a system, such as the above described, might be useful in a real context, a prototype application has been developed, which intends to provide the users of a public transport with a better view of the landscape outside the vehicle.

The specific goals of this dissertation are:

- Study the needed technologies for developing a GPS plus Computer Vision (CV-GPS) system;

- Describe in detail the CV-GPS system, as well as an application example where that system is needed;
- Implement a prototype of both the referred application, and the CV-GPS system;
- Test the CV-GPS system and discuss the obtained results.

Summarizing, the practical and final result of this dissertation should demonstrate that a combined approach of GPS and Computer Vision can improve geolocation. For this demonstration will be implemented a prototype of the referred system, as well as an application that needs that system for geolocation. That application has the purpose of allowing the user to have a clear view of the outside landscape in a mobile device such as a smartphone or tablet, by streaming a captured video of the landscape and complementing it with textual information about a set of points of interest, known by the system.

1.4 Research Question

The main research question trying to be answered is the following:

- Can the actual geolocation be improved by integrating a computer vision component that analyses captured frames of the landscape, trying to find reference points in the image and making them correspond to a determined location?

1.5 Document Structure

This document is structured as follows:

Chapter 1 - Introduces the topic of this dissertation, as well as describes the motivation and objectives of this work.

Chapter 2 - Present the revision of the state of the art in the fields of research related to this work. This chapter is divided according to the technologies we intend to explore and is finalized by a subsection of related applications.

Chapter 3 - Describes the main system of this thesis, introducing use cases with problems that the proposed approach intends to solve and presenting the general solution (without implementation details).

Chapter 4 - Presents a more technical view of the solution describing all the elements that are part of the developed system as well as the prototype implementation.

Chapter 5 - Describes the tests applied to the prototype and presents the obtained results, discussing and comparing them with the expected results.

Chapter 6 - Presents the conclusion of this work, ending with a description of some directions of future work.

Chapter 2

State of the art

This chapter revises the existing technologies related to the research subject and evidence some references and projects related with those technologies. The following systems/technologies are described in the next sections:

- Geographic Location on Mobile Devices;
- Computer Vision.

2.1 Geographic Location on Mobile Devices

This section explains which methods and devices can be used to obtain an accurate position of a mobile device. There are three main technologies available that can easily return the position of a device:

- Global Positioning System;
- Cell Tower ID;
- Assisted-GPS.

2.1.1 Global Positioning System

The Global Positioning System, also known as GPS allows every person/device to access its exact location by using a GPS receiver. The GPS provides an accurate measure of the latitude, longitude and altitude of any place on Earth, being composed by three segments: the spacial segment, the control segment, and the user segment.

The first segment consists in a set of 24 satellites distributed in 6 orbital planes equally spaced at an approximate altitude of 20.200 km. All the satellites are placed so that at least 4 of them are always visible on any place of the Earth surface.

The second segment is the control segment, which is responsible by:

- Monitor and control the satellites system;

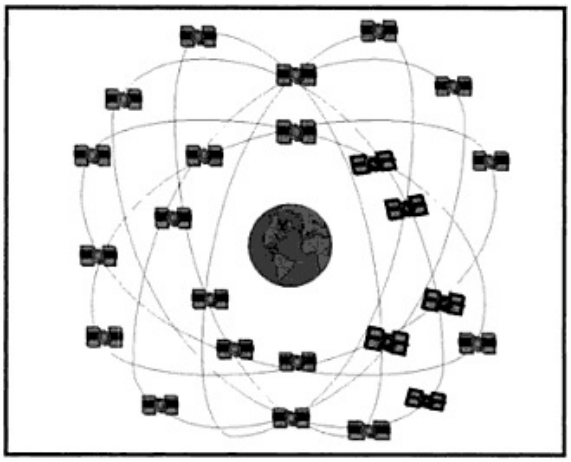

Figure 2.1: Satellites Constellation (retrieved from [Mon00])

- Determine the GPS system time;
- Calculate the time corrections on the satellites;
- Periodically update the navigation messages of every satellite.

This very important segment is composed by a set of monitoring stations, which guarantee that any deviation in the satellites route is promptly fixed.

The third segment is the user segment, composed by the GPS receptors, capable of reading the GPS messages send by the satellites. All the receivers have at least 12 channels, meaning that they can simultaneously receive and process the signals of at least 12 satellites. Earlier instruments had to process data in series, making them considerably slower and less accurate and more sensitive against disturbances. Todays receivers can follow up to 66 GPS signals (more than are currently available) and deal with multipath and reflected signals. Some of them are also sufficiently sensitive to work indoors [Kon09].

This receptors can be separated in two major categories: civil and military. Military uses GPS receptors, in many cases, to track soldiers or to guide missiles to their target. Civilians use them, for example, on navigation systems, to find the best path between two coordinates. [Mon00]

Figure 2.2 illustrates a GPS navigation system developed by NDrive Navigation Systems.

2.1.1.1 GPS Precision

Normal GPS receivers always have an associated error. Measuring this error when the vehicle is moving at high velocity is not easy, because it is necessary to know the exact position of the vehicle in real time and to make the comparison with the values returned by the GPS, but measure its precision at a well known place, is possible. GPS sensors are relatively accurate when a set of satellites are fixed (the position of the satellites is known) and the receiver is not moving. With open sky, the measure accuracy is 5,3 meters ([WEK05]) but when under heavy canopy (under

Figure 2.2: Mobile GPS Application (retrieved from [NDr12])

dense forest) the GPS receivers have an average error of approximately 7.6 meters with a standard deviation of approximately 4 meters ([WE07]). At an urban environment this errors can increase, because the GPS radio signal is highly affected by massive metal structures. This way, although GPS precision can sometimes be extremely accurate, even with low-cost receivers, that accuracy tends to decrease if the receiver has not a clear path to the satellites or is moving at high velocity.

2.1.2 Cell Tower ID

Mobile devices usually support different connection signals like Third Generation (3G) or Fourth Generation (4G), wireless networks, bluetooth or GPS. Smartphones always use, at least, the cellular network to communicate, sometimes having GPS receivers too. In tablets it is more common the existence of a GPS reveiver, wireless networks or even bluetooth, but not all have cellular networks.

When a mobile device is using cellular networks, it needs to be connected to at least one cell tower, knowing the cell ID of that tower. This way, it is possible to get the devices position, even if the GPS is not available or turned-off, using one simple network based (GPS-free) positioning method. This method is based in cell coverage, evaluating the cell identification and is commonly deployed by mobile network operators. The position of a mobile connected to a particular antenna, which is identified by its cell ID, is determined by the location of the base station itself. A more advanced network-based GPS-free positioning method using cell coverage, determines the location of a mobile connected to a multiantenna station by evaluating the center of gravity of the sector that the mobile belongs to, thus using that location as the estimated position of the device. [FF10]

Google has a system that provides a similar service. When a mobile device with GPS sensor is connected to a cell tower, it sends its accurate position plus its current antenna cell ID and cell coverage to this Google service. With this information, Google knows with a controlled accuracy the cell tower's location. This way, when a user without GPS queries the service, Google translates the signal power and the provided cell ID into geographic location. This method can be a good solution when the users device has no GPS signal, such as happens, for example, inside buildings. [Goo08]

2.1.3 Assisted-GPS

Assisted GPS (A-GPS) improves standard GPS performance by providing data that under normal conditions the GPS receiver would receive from the satellites, through an alternative communication channel. Figures 2.3 and 2.4 show overviews of an A-GPS system. Note that A-GPS does not excuse the receiver from receiving and processing signals from the satellites; it simply makes this task easier and minimizes the amount of time and information required from the satellites.

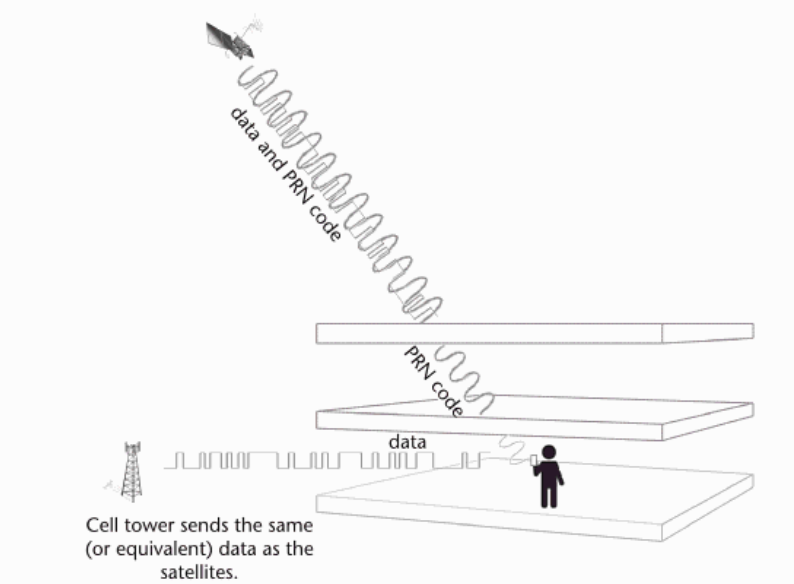

Figure 2.3: A-GPS (retrieved from [Dig09])

Each GPS satellite sends a pseudorandom noise (PRN) code, as well as a data stream. The PRN code is illustrated in the diagram by a sinusoid and the data is illustrated by a square wave. As the signal moves through obstructions it gets weaker; the data may not be detectable, but the code still is. In an A-GPS system the same, or equivalent data is provided via a cell tower. The A-GPS receiver receives the same information that it could have obtained from the satellite if the signal was not blocked. The same concept also allows the A-GPS receiver to compute a position quicker, even if the satellite signal is not blocked, because the data can be sent much faster from the cell tower than from the satellite.

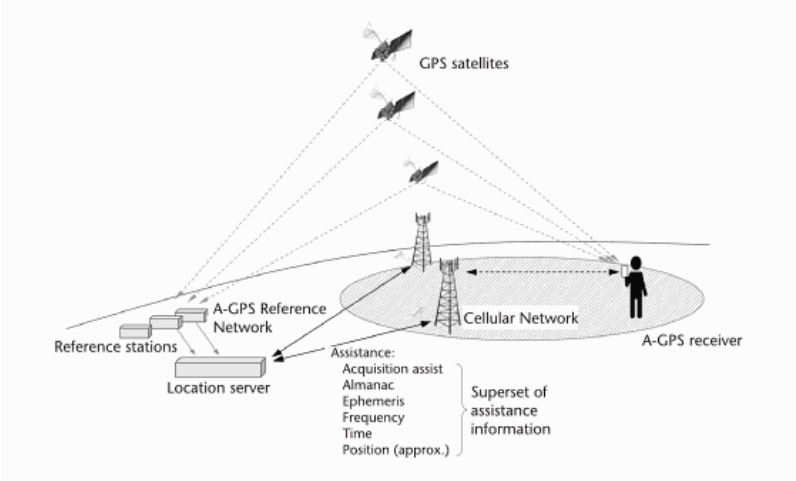

Figure 2.4: A-GPS System(retrieved from [Dig09])

The satellite data is collected and processed by an A-GPS reference network and location server. The assistance data is usually provided by a cellular data channel. The approximate position of the A-GPS receiver is usually derived from a database of cell tower locations. [Dig09]

2.2 Computer Vision

"If we think about some of nowadays hottest topics in Computer Vision, such as image retrieval or object recognition, it is almost impossible not to think about using features. Features can be thought as significant properties of an object than can be used as part of input to processes that lead to distinguish the object from other objects or to recognize it. Thus, it is important to know which parts of the object are relevant and characteristic of it, and then describe them in a way that we can enhance the differentiable properties." [BVnS10]

As stated by Bernal, when thinking about object recognition it is almost impossible not to think about using feature recognition. Objects have characteristic properties that distinguish them like texture, color or shape. By detecting some of that features, it is possible to detect an object or distinguish it from another. In this chapter, feature detection will be introduced, and some feature descriptors will be described, in order to provide a consistent study about this topic.

2.2.1 Feature Detection

Feature detection is an important part of Computer Vision algorithms. The goal of feature detection, is to obtain feature descriptors, locating points and regions in an image. The problem with feature detection, is that the concept of "feature" varies accordingly the context where it is applied. For instance, if the objective of the application is the detection of emotions through a picture of a human face, the features to detect will be kind of different from the features detected in building's recognition.

There is no universal definition of what constitutes a feature, so it depends of the context where that features are recognized. Feature detection is often a low-level image processing operation, and it can be divided in four main groups: Edge Detectors, Corner Detectors, Blob Detectors and Region Detectors. [BVnS10]

In Figure 2.5 it is possible to find an overview of these groups.

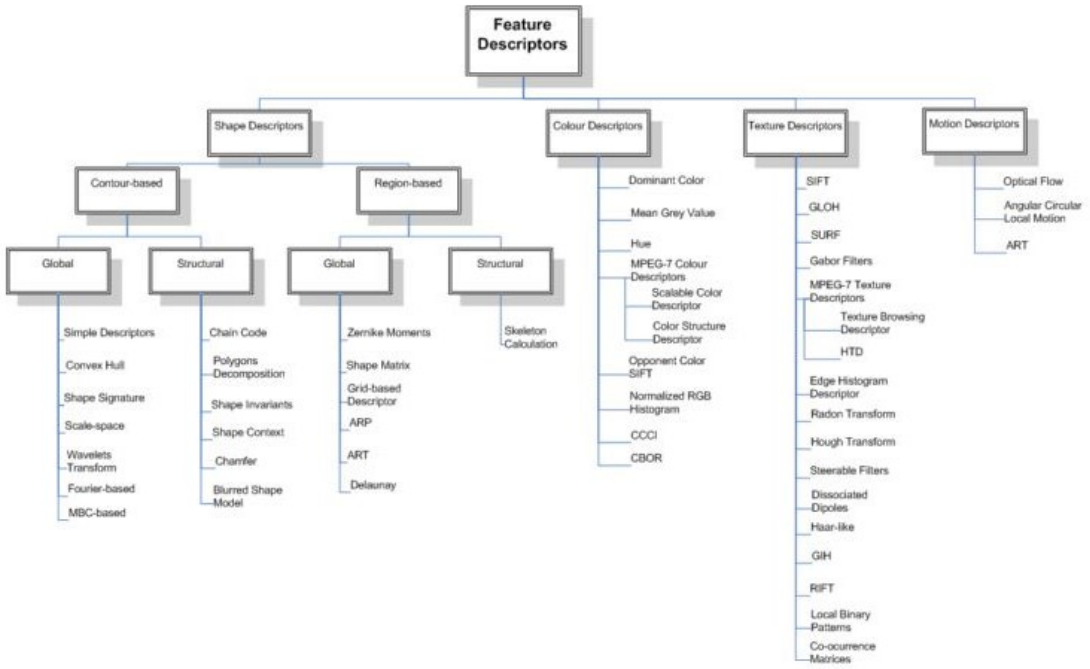

Figure 2.5: Classification of Feature Detectors (retrieved from [BVnS10]).

The most common algorithms are edge and corner detectors. Region detectors are mostly used on object tracking.

Accordingly to Bernal, Feature Descriptors can be separated by texture, color, shape or motion [BVnS10]. To obtain good results in open sky object recognition, where the object's colour can vary depending of the illumination, it is important to give greater focus to texture descriptors instead of colour descriptors.

2.2.2 Feature Descriptors

Nowadays it is possible to find countless feature descriptors, adapted to many different contexts. In this subsection some of the most used texture and shape descriptors are presented and compared.

2.2.2.1 Scale Invariant Feature Transform

The Scale Invariant Feature Transform (SIFT) algorithm was presented in 1999 by David Lowe.

SIFT transforms an image into a collection of feature vectors, each of which is invariant to image translation, scaling, and rotation, and partially invariant to illumination changes. The SIFT features share some properties with the neuron's responses, being less sensitive to projective distortion and illumination change.

The first stage of the algorithm identifies key locations in scale space by looking for locations that are maximum or minimum of a Gaussian function. Each point is used to generate a feature vector that describes the local image region sampled relative to its scale-space coordinate frame. The features achieve partial invariance to local variations, such as affine or 3D projections, by blurring image gradient locations.

The SIFT keys obtained from an image are used in a nearest-neighbour approach to identify candidate object models. Using a Hough transform hash table is possible to first identify the collections of keys that match with a potential model, and using a least-squares fit it is possible to obtain a final estimate of model parameters. There is strong evidence for the presence of the object in an image, when at least 3 keys agree with the model parameters. Since there may be dozens of SIFT keys in the image of a typical object, it is possible to have substantial levels of occlusion in the image and yet obtain high levels of reliability.

For instance, on an example application developed by David Lowe in 1999, each image generated 1000 SIFT keys, in a process that only required less than 1 second of computation time. [Low99]

SIFT algorithm is the base for many others that have extended it. Two examples are the following presented algorithms, Histogram of Oriented Gradients and Speeded-up Robust Features.

2.2.2.2 Histogram of Oriented Gradients

Histogram of Oriented Gradients (HOG) is implemented by dividing the image window into small cells, for each cell accumulating a local one dimensional histogram of gradient directions or edge orientations over the pixels of the cell. To improve this method quality it is possible and useful to contrast-normalize the local responses before using them, accumulating a measure of local histogram energy over larger blocks and using the result to normalize all the cells of that block. This will bring better invariance to illumination and shadowing. Overlapping the detection window with the grid of histogram oriented gradient above described and using the combined feature vector gives our human detection chain. [DT05]

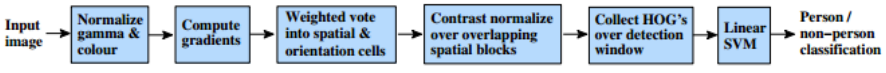

Figure 2.6: Histogram of Oriented Gradients process (retrieved from [DT05]).

The detector window is tiled with a grid of overlapping blocks in which Histogram of Oriented Gradient feature vectors are extracted. The combined vectors are fed to a linear support vector machine (SVM) for object/non-object classification. The detection window is scanned across the image at all positions and scales, and conventional non-maximum suppression is run on the output pyramid to detect object instances. [DT05]

The use of orientation histograms already occurred before HOG, but it only reached maturity when combined with local spatial histogramming and normalization in Lowe's Scale Invariant Feature Transformation (described in the previous chapter) approach to wide baseline image matching. The success of sparse feature based representations has somewhat overshadowed the power and simplicity of HOG's as dense image descriptors.

The HOG representation has several advantages. It captures edge or gradient structure that is very characteristic of local shape, and it does so in a local representation with an easily controllable degree of invariance to local geometric and photometric transformations: translations or rotations make little difference if they are much smaller that the local spatial or orientation bin size. [DT05]

2.2.2.3 Speeded-up Robust Features

Speeded-up Robust Features (SURF) descriptor is based of SIFT although it reduces its computation time. SURF detector works in a similar way as SIFT but it has some differences, being most important the use of integral images to reduce the computation time. Integral image is an algorithm for quick and efficient generation of the sum of values in a rectangular subset of a grid, where the value at any point (x; y) in the summed area table is just the sum of all the pixels above and to the left of (x; y). SURF is a Fast-Hessian Detector, because it is based on the calculation of a Hessian determinant.

As with SIFT, Gaussian filters are optimal for scale-space analysis but in practice, Gaussian implies a discretization and crop of the image, where aliasing effects can appear, so it seems that the use of Gaussian filters could not be the best option. To solve this problem the SURF author suggest the use of a box filter that use second order Gaussian derivatives, making this process much faster. The 9x9 boxes are approximations for Gaussian second order derivatives. [BTV06]

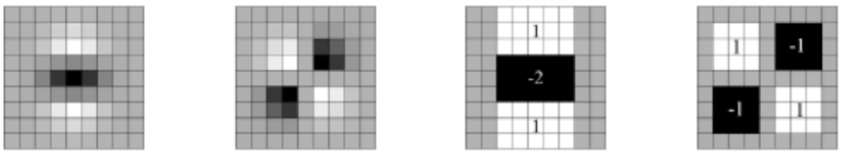

Figure 2.7: Discretized and cropped Gaussian second order derivative approximations using box filters. The grey regions are equal to zero. (retrieved from [BTV06]).

The images are also repeatedly smoothed with a Gaussian and subsequently sub-sampled. Thus, by using box filters and integral images it is not necessary to apply the same filter to the output of a filtered layer but being possible to apply the same filters directly in the original image,

even in parallel (which reduces computation time). The Scale-space is analysed by up-scaling the filter size rather than iteratively reducing the image size. The output of the first layer of filters is considered as the initial scale layer. The following layers are obtained by filtering the image with gradually bigger masks (9x9, 15x15, 21x21, etc). [BTV06]

SURF implementations have accurate results even when colours and viewpoints are different as we can see in Figure 2.8.

Figure 2.8: The same object viewed from different viewpoints and with different colours (retrieved from [BTV06]).

2.2.2.4 ORB

The Oriented-FAST and Rotated Brief (ORB) [RRKB11] is a recent descriptor based, as the name implies, in FAST and Brief detectors. The objective of this descriptor is eliminate several disadvantages of both FAST and Brief algorithms and exploit the good performance and low-cost they have.

FAST [RD06] and its variants are the method of choice of keypoints in real-time, but don not include an orientation operator. Many of these operators involve histograms of gradient computations or the approximation by block patterns, but both methods are either computationally demanding or present bad results.

Brief [CLSF10] is a recent feature descriptor, similar do SIFT in many aspects like robustness to lightning or perspective distortion but with a poor performance in rotation.

The main contributions of the ORB detector is mainly to eliminate the difficulties of Fast and Brief, through the addiction of a fast and accurate orientation component to FAST and an efficient computation of BRIEF features.

Accordingly with the evaluation tests performed by the authors of this feature detector, ORB is capable of detecting more characteristic points and considerably faster than its competitors, being considerable superior to the other most used feature detectors: SURF and SIFT [RRKB11].

2.2.2.5 Other Texture Feature Detectors

The table bellow, 2.1, present a brief overview of relevant texture feature descriptors, comparing them for rotation invariance, translation invariance and scale invariance.

Feature Descriptor	Basis	Rotation Invariance	Translation Invariance	Scale Invariance
SIFT	Describing a patch of an image by pondered oriented gradient histograms.	Yes	No	Yes
SURF	Similar to SIFT but uses integral images	No	No	Yes
Gabor Filters	Bank of filters with different dilations and rotations.	No	Yes	No
Texture Browsing Descriptor	Characterization of the texture's regularity, directionality and depth.	No	Yes	No
HTD	Find homogeneous texture pattern along a patch to help image retrieval.	Yes	Yes	No
Edge Histogram Descriptor	Captures edges spatial distribution.	No	Yes	Possible
Radon-based Descriptor	Converts the original pixel represented images to Radon-pixel images.	No	No	No
Hough Transform Descriptor	Useful to fit some special shapes in a image such as lines, circles under a statistical approach.	No	Yes	No
Disociated Dipoles	Comparison off the mean illuminance value of two regions.	No	No	No
Steerable Filters	Finds out the response of a filter in several orientations without having to calculate them all.	No	Yes	No
GIH	Keeps information of the joint distribution of the geodesic distance and the intensity of the sampled points.	No	No	No
RIFT	Calculates Histogram of Oriented Gradients for each ring.	Possible	No	No
Local Binary Patterns	Description of each pixel by the relative grey levels of its neighbours.	No	Yes	No
Co-occurrence Matrices	Frequency of each pixel value's following another possible value appearence.	No	Yes	No

Table 2.1: Summary of Texture Descriptors (retrieved from [BVnS10]).

The information presented in the table indicates that two algorithms distinguish from the others by being scale invariant: SIFT and SURF. Scale invariance is an important factor when the objects to detect may be displayed in different sizes in the image. Besides this characteristic, SIFT has the advantage also support rotation invariance, which is important in scenarios where the object is displayed on different angles in the image. The invariances presented in the figures may sometimes be important in the detection, but is necessary to consider that sometimes the detection

time is increased due to this features. In the choice of a feature detector must be considered which invariances are most needed, and which implications they have in the detection time taken.

2.3 Application Examples

This section presents some example applications regarding, separately, the two technologies previously described.

2.3.1 Geographic location

2.3.1.1 Google Maps

Google Maps is a navigation system developed by Google that uses the device GPS or A-GPS system to find its position. Besides GPS navigation, one of the main functionalities of Google Maps is the My Location system that may work offline if the user has his GPS turned off, by using the Google Cell Tower ID system.

The figure 2.9 shows 3 screenshots of the Google Maps application.

Figure 2.9: Google Maps example (retrieved from [Goo]).

2.3.1.2 GPS Status

GPS Status is a mobile application that reads all the geolocation sensors available on a smartphone or tablet, returning information as the number of satellites visible and fixed, the latitude, longitude and altitude coordinates, the estimated speed of the device, and an error estimate of this readings. One screenshot of the application can be found in Figure 2.10.

Figure 2.10: GPS Status example (retrieved from [Mob]).

2.3.2 Computer Vision

2.3.2.1 Google Driverless Car

Google Driverless Car is being developed by Google towards the development of an autonomous vehicle. The project is currently under development, and it is already authorized to work in the public roads of Nevada, United States of America [Reu12]. This car uses many sensors like laser, high resolution cameras, radars, GPS, wheel encoders, etc, to get accurate representations of the real world [Spe11]. Figure 2.11 shows the first driverless car developed by google, an adapted Toyota Prius, and Figure 2.12 shows the generated model that represents the car's "vision".

Figure 2.11: Google driverless car example (retrieved from [Spe11]).

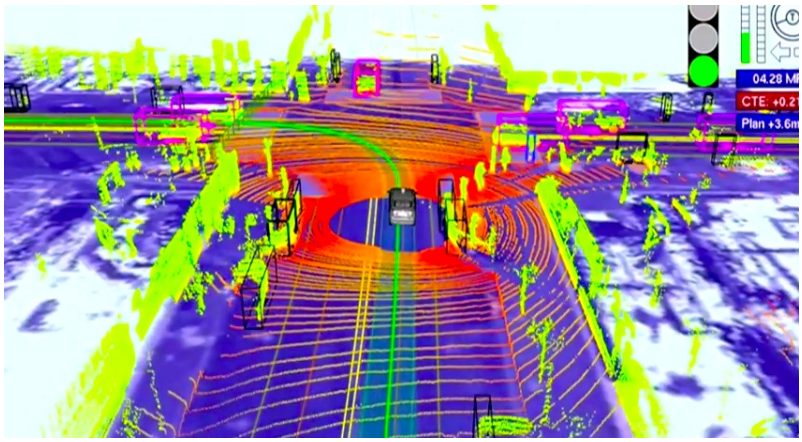

Figure 2.12: Google driverless car vision example, generated by the data retrieved from all of its sensors (retrieved from [Spe11]).

2.3.3 Overview

The application examples presented in this chapter, use separately both the enabling technologies necessary for the development of this dissertation. Although there is not an application example using both technologies with the same goal of this dissertation, it is possible to conclude in which context each technology fits, and what is their utility for this work.

The Google Driveless Car, for instance, uses GPS sensor and computer vision (between other sensors), to get a mapping of the real world and take decisions about what directions to follow. The conclusions that this work is trying to achieve are much simpler, but intend to use the same technologies, namely GPS and Computer Vision, to obtain results for a different context. Next chapter introduces a possible solution for the geolocation problem described in the introductory chapter.

Chapter 3

Improving geolocation accuracy by combining GPS with Image Analysis

Following the background research, this chapter specifies the Computer Vision plus GPS system, designated CV-GPS system. This chapter describes an application scenario that was the origin for this work, presenting application examples for the CV-GPS system, that can not be solved by any other geolocation technology described in the previous chapter. Still in this chapter, the operation of the CV-GPS system is explained in detail, as well as an architecture overview of the application scenario.

3.1 Application Scenario

The application SVI_Trans (Scenery Visualization in Transport Vehicles) was developed for Fraunhofer AICOS Portugal, a research center that focuses in the improvement of the accessibility and usability of the information and communication technologies(ICT) with the target of improving the seniors life quality. The SVI_Trans application fits this objectives by trying to solve a public transports's problem that is the lack of clear vision and touristic information that a public transport's passenger has inside the vehicle and during a travel.

When a passenger is travelling on a public transport vehicle, for example in a bus, his vision is conditioned by the structure of the vehicle itself, by advertisement in the windows, or even by other people standing in the vehicle. In an airplane, the small size of the windows is critical and prevents the user from having a clear view of the outside.

This problem highly affects the quality of the travel, preventing the passenger of knowing the places he is passing by and changing what could be a pleasant and interesting travel in a boring trip where the time is not fully enjoyed. When someone is travelling through a city, wants to see buildings, landscapes, and every point of interest (POI) that the city might have, and possibly find more information about each POI. Similarly, when someone is travelling in an airplane, may be

interested in seeing more information about the locations that his vision is capable to reach, discriminating the rivers, mountains or cities in the landscape.

The problem described in the previous paragraph, concretely the difficulty of visualization, from someone inside some public transport vehicle, to the outside, and the lack of touristic information about the existing POIs in the route of that public transport vehicle, may be solved with the help of a mobile application, capable of reproducing a real-time video of the landscape and of knowing when the vehicle is getting closer to a POI.

With the recent advances of the technology and the increased processing capacities of the mobile devices, it is possible to develop such an application and help the public transport's passengers to have a better travel. However, with the actually known geolocation systems, described in section (2.1), namely GPS (2.1.1), Cell Tower ID (2.1.2) and A-GPS (2.1.3), it is difficult to accurately know when the vehicle is approaching some POI.

As previously said, GPS is affected by the lack of clear view to the satellites, massive metal structures or high velocities. The A-GPS tries to resolve GPS problems using auxiliary information, received from remote servers using a network connection, but this connection may not be available everywhere. Cell Tower ID needs network connection too, and its accuracy varies from point to point, reason why its only used when there is no GPS receiver available.

In a hypothetical scenario, the described application is implemented in a bus, using only GPS to reference its position. The bus is travelling in an urban environment, following a defined route, and suddenly enters a long road in the middle of many skyscrapers, as in Figure 3.1. In this situation, the GPS signal could be lost, because the receiver would not have direct view to the satellites, and the system would automatically fail due to not knowing its location. Even with the use of A-GPS instead of GPS only, the system could fail, because the network connection would be affected by the massive metal structures surrounding the receptor.

The lack of a reliable geolocation technology, led to the need of a new geolocation system, capable of accurately knowing that the vehicle is approaching some POI if the GPS information fails. Using computer vision (2.2) to complement GPS information, it is possible to detect reference points on an image, and associate them to a known location. This way, in the previous hypothetical scenario, the lack of GPS information would trigger an image analysis system, that would begin to seek for reference points in a set of captured frames. If any positive match is found, it is possible to conclude not only that the vehicle is close to that reference point, but that it is still behind it. This would keep the system working and knowing its relative position, even without GPS information.

In a second hypothetical scenario, the described application is implemented in a bus, using only GPS to reference its position. The bus is travelling in an urban environment, following a defined route, and approaches some POI. At some point, the bus is 80 meters behind that POI.

Figure 3.1: Urban Canyon example.

In this situation, the error associated to a GPS receiver, which may vary in worst cases to values around the 100 meters, may indicate that the vehicle is in any position between 180 meters behind the POI, to 20 meters ahead the POI. This lack of accuracy, would have consequences on the users application, which would start displaying information about the POI either too soon, or too late, referencing a POI that is not yet visible to the user or that was already passed by the vehicle.

By combining GPS information and computer vision, the application would properly work in the second hypothetical scenario, because the image analysis system would be triggered at a considerable distance of the POI, for instance 300 meters [1], this way guaranteeing that the vehicle did not yet passed by it, and when a match is found between the real-time captured frames and the POI images saved in a database, the system knows that it is getting closer to that POI, but did not yet passed by it. This way, the information would be properly displayed to the user.

This two hypothetical scenarios are good examples to demonstrate the utility of the CV-GPS system. Moreover, this system would be useful in other situations such as, for example, vehicles moving at a considerable velocity, where the GPS positioning processing might suffer delays.

In the next section the proposed geolocation system, combining GPS and computer vision, is presented and fully explained.

[1] Distance used in the implementation.

3.2 CV-GPS System

The idea to resolve the problems mentioned in the previous section and in the introductory chapter (1.2) combine two distinct technologies: GPS (2.1.1) and Computer Vision (2.2).

If used separately, each of these technologies presents problems. GPS alone can not provide an accurate positioning in situations like the described in the previous chapter, where there is no GPS signal or the existing signal is weak, and computer vision itself can not be used to reference a location, because with the current processing capacity it is impossible to execute a real-time matching between millions of images (it is necessary a reduction of the set of images to compare). This way, although distinct, these technologies may complement each other, in order to create a valuable new system, able to compensate the complete or partial lack of positioning information.

The main idea of this system is to receive and analyse the GPS coordinates sent by a GPS receiver, triggering the image analysis if the GPS signal is absent or weak. To achieve that, it is necessary to combine both technologies, making an image correspond to a location. This correspondence may be accomplished with a database of images and coordinates, where an image of some reference point is related with the respective geolocation coordinates. A reference point is an item kept in a database, with GPS coordinates and images associated to it. An example of a reference point, might be a building.

The operation of this system, is better explained in Figure 3.2. Considering X as the limit distance in meters, from which the image analysis may start being executed, and Y as the limit time in milliseconds that the system may be absent of GPS information before starting the image analysis, is possible to better understand when the image analysis it is triggered.

The system needs to be ready to receive and process the GPS information sent from a GPS sensor. If that information is received, then latitude and longitude coordinates are parsed from it. If the GPS sensor may read its own accuracy (value in meters correspondent to the approximate error of the coordinates), then accuracy information is as well sent to the system, to be analysed. In this case, if the GPS information is accurate enough for the geolocation, then the latitude and longitude coordinates may be used to calculate the distance from the current position to the closest set of reference points. This calculation has the purpose of discarding the more distant reference points, reducing the number of valid points for analysis, at a given moment. When the distance to the closest reference point is smaller than a threshold value X, the image analysis is started, guaranteeing that the error associated to the GPS information will not affect the geolocation results. If, on the other hand, the received GPS information is not accurate enough to be reliable, then the image analysis is started, to compensate that lack of valid geolocation information. If the system suddenly stops receiving GPS information, a timer is started and incremented, until GPS information is available again or a maximum time Y is exceeded, in which case the image analysis is also triggered.

In all the above presented scenarios, where the image analysis is started, if the result of that

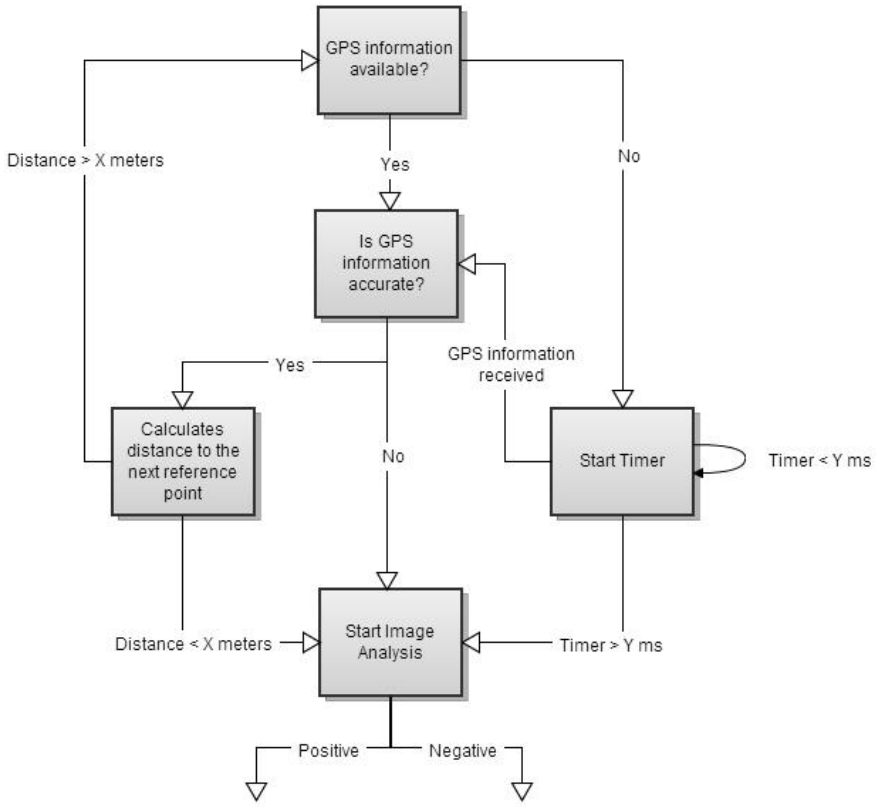

Figure 3.2: CV-GPS system operation.

analysis is positive, the system may conclude that it is close to the geographic coordinates associated to the matched reference point.

After this overall description, it is important to detail the image analysis operation. Every time the image analysis is processed, it requests images from two sources. One source is an external camera capturing images in real time (observed image). The other source is a database of reference points (model images). From the database, are only considered images of the points closer to the actual location. In order to obtain good performances, the number of image comparisons should be the lower possible, avoiding unnecessary image analysis. To analyse the images, a feature descriptor is used (2.2), which as the name implies, detects characteristic elements (features) between two images and compares them trying to find similarities. To guarantee better results, it is important that every reference point has more than one associated image, from distinct points of view. Ideally, to every reference point should correspond at least three images [2], one frontal and two lateral, in which case only one match would be necessary to obtain a positive result.

The described approach may effectively improve the accuracy of geolocation, compensating the GPS in critical situations, as it will be demonstrated in the evaluation chapter (5). Next section

[2]Number of points of view of the building from the road (one frontal view and two lateral views)

presents an architecture overview that explains the necessary segments to implement this system.

3.3 Architecture Overview

Conceptually, the application has four main segments: media, server, GPS receiver, and the client application. Figure 3.3 illustrates the general architecture of the system.

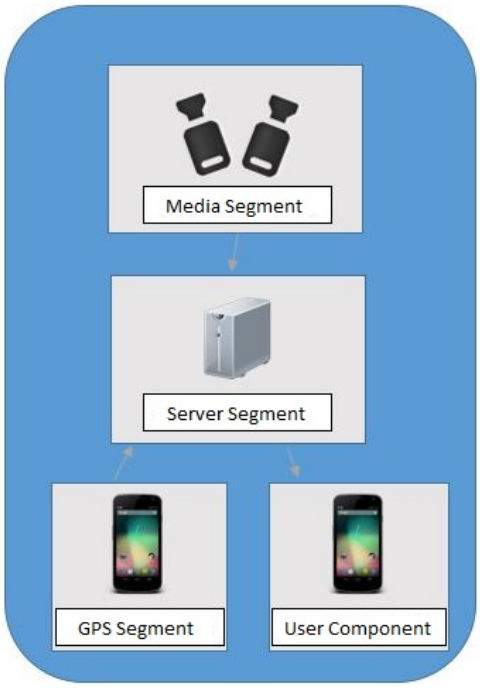

Figure 3.3: General architecture of the system

The **Media segment** is composed by a camera (or set of cameras), properly configured to provide access to one or more video streams. It is important that these cameras are strategically placed in the vehicle, in order to obtain a clear view of the outside landscape. If this segment is composed by only one camera, it should be positioned in the front of the vehicle and pointing forward, so the requested images may display a reference point before passing by it. If this segment is composed by more than one camera, than only one of these cameras must be placed as above described, and the others may be placed freely.

The **Server segment** is responsible for most of the processing activity, receiving the video streams from the media segment, and the GPS information from the GPS segment, and processing the information in order to obtain a valid and accurate positioning of the system. This segment incorporates important functionalities like Streaming, Recording or Loading where, parallel to the analysis of the information previously described, a set of processing activities are running, providing the video and the touristic information to the user segment. The CV-GPS system is an important part of this component.

The **GPS segment** is composed by only one GPS receiver. This receiver must be correctly positioned and fixed in the vehicle, in order to guarantee that the lack of accuracy of GPS information is only resultant of the signal reception itself, and not from the bad use of the receiver. By properly fixing the receiver in the vehicle it is possible to associate the received coordinates not only to the GPS receiver, but to the vehicle itself. Besides the signal reception, this segment must be capable of send the information to the server segment, where it is properly analysed.

The **Client segment** consists in a mobile application where public transport's passengers can see a real-time video stream of the exterior landscape, and access touristic informations like names or descriptions about some POIs. This component must be the most intuitive and simpler possible, in order to provide the user a good travel experience, and to prevent the waste of time learning how to use it. On the other hand, it is important that this application may be customized for different ages (personalizing factor like for instance the letter size).

3.4 Summary

In the current chapter, the Computer Vision plus GPS solution was presented, and a few use cases were described, demonstrating the utility of this system when integrated in a real application scenario. Given this considerations, next chapter presents the implementation of the CV-GPS as well as the implementation of the already presented SVI_Trans application prototype.

Chapter 4

Implementation

The system presented in the previous chapter was implemented following the architecture overview presented in section 3.3. This chapter describes in detail the implementation of that system and of the developed prototype, explaining the implementation decisions taken until the development of the final version of the prototype. The goal of having a prototype is to test the hypothesis presented in the chapter 1 and further described, and to take conclusions about the utility or not, of the CV-GPS system.

4.1 Architecture

This section describes the architecture of the implemented system, detailing every segment's architecture in order to provide a precise vision of each one. In this development were used many technologies, explained and justified in every segment description.

4.1.1 Media segment

The media segment is composed by a set of IP-based network cameras, configured to diffuse video to the server segment. These cameras have two different access protocols: Real Time Streaming Protocol (RTSP [SRL98]) or Hypertext Transfer Protocol (HTTP). Both protocols allow video broadcasting, which means that the stream may be accessed from several points simultaneously. Associated to these protocols are two different video encodings: H.264 and Motion JPEG.

H.264 ([WSBL03]) is a compression pattern often used on recording, compression and distribution of high definition video, and Motion JPEG is a video format often associated to digital cameras, IP cameras or webcams, which offer some advantages like performance or support.

Motion JPEG ([FS95]) is supported by many applications, inclusively for some web browsers or mobile video decoders, which makes it good for applications that need great portability. H.264 does not have the same level of support, not being compatible with web browsers like Google Chrome and being only partially supported by Android. On the other hand, H.264 can provide better video quality, by using a higher video bitrate than Motion JPEG.

The information flow in this segment is illustrated in Figure 4.1, which specifies the inputs and outputs of the media.

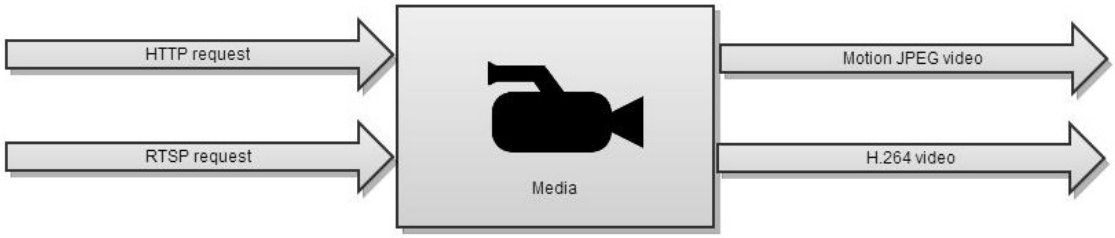

Figure 4.1: Information flow on the media segment

4.1.2 Server Segment

The server is the main segment of the system, performing the management of the video contents and processing the GPS information. Figure 4.2 describes the internal architecture of the server.

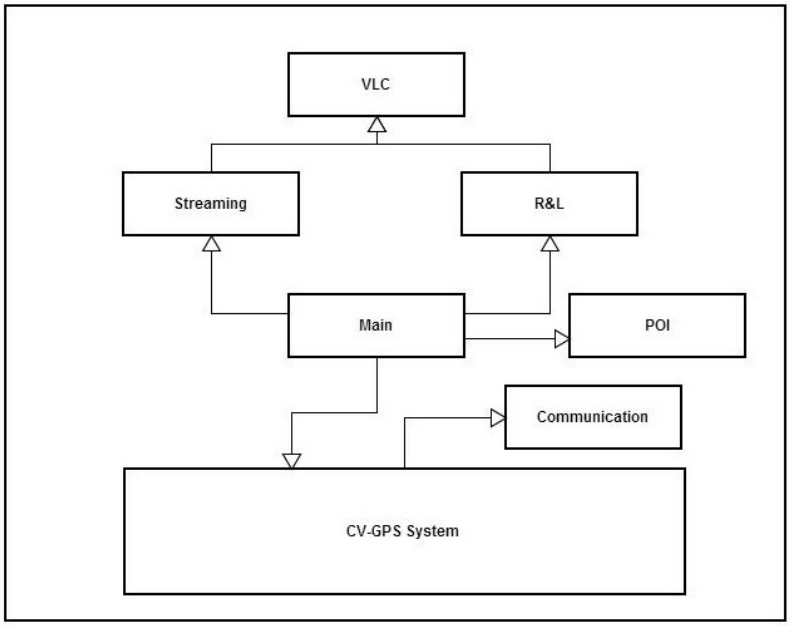

Figure 4.2: Architecture of the Server segment. R&L corresponds to the Record and Load module, DM to the Decision Module and IA to the Image Analysis module.

As illustrated in Figure 4.2, the server is composed by a set of modules responsible by most of the functionalities of the application. These modules are:

- Main module, responsible by initialize the system;
- Streaming module, responsible by the video streaming;
- Record and Load modules (R&L) responsible by saving the received information and replicate it when needed;

- POI module, responsible by manage the POI information;
- Communication module, responsible by the communication with external applications;
- CV-GPS module, responsible by the analysis of GPS and CV information, in order to obtain a more accurate geolocation.

The modules above described interact with each other to perform the server activity. The system can perform three operating modes: Streaming, Recording and Loading, further explained in the prototype implementation (4.3). The **Main module** initializes the remaining modules, accordingly with the operating mode chosen by the user.

The **Streaming module** performs only operations related with the video stream. The video management is performed by an external application: VLC Media Player. VLC is a media player of multiple capabilities that allows the user to capture, save and stream video in several resolutions and encodings, and that may be run from a console interface (making its integration with the system much simpler).

The **Record and Load module** has a similar performance to the Streaming module, but instead of stream the video and the information to the user segment, it saves the information in a database and configures it to be loaded, when requested.

The **Points of Interest** (POI) module initializes the points of interest, correspondent in this scenario to historic or characteristic buildings, in a structure well known by the application. For each POI it is necessary to know the GPS latitude, longitude and accuracy, the name and small description (that is shown in the mobile application when a public transport is passing by the point of interest), a link where is possible to find more information about that point, and the paths to the model images of the point. Part of this information is used in the CV-GPS module (GPS information mainly), and another part of it is sent to the clients mobile application (name, description and link for more information).

The **Communication module** is the module responsible by communicating with the user segment. If requested by the CV-GPS module, it sends information about some POI to all the client applications connected and running.

The more important part of this server, and the core of all application corresponds to the CV-GPS module. This module implementation is explained in detail in the next section, 4.1.2.1.

4.1.2.1 The CV-GPS architecture

As referred in chapter 3, this module is based in two different types of information: GPS (latitude, longitude, accuracy) and Image Analysis.

The implemented CV-GPS has three different inner modules: the Decision Module (DM), the Computer Vision module (CV) and the Image Analysis module (IA). Figure 4.3, illustrates the module architecture.

The **DM** is responsible for the decision making, considering and analysing the information above described in order to make a valid decision about the current geolocation of the system.

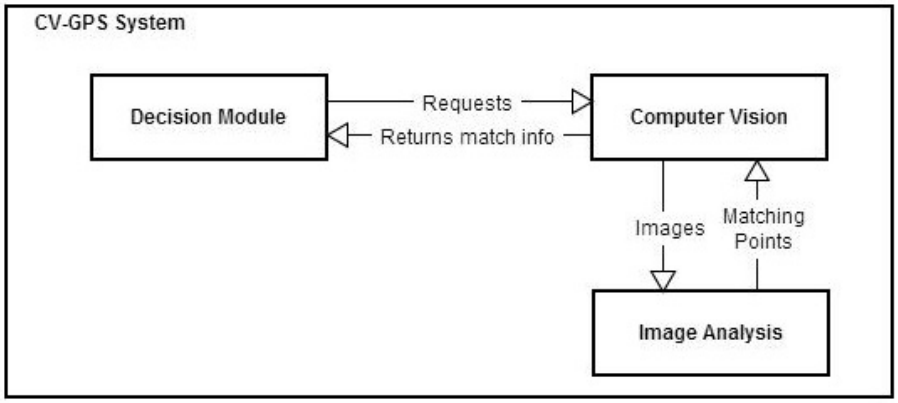

Figure 4.3: CV-GPS system architecture.

The **CV** module is responsible for the management of the image analysis. This module requests a real-time frame from the media segment and loads the images of the reference points that will be analysed. When all the images are available, it starts launching new IA modules, comparing the captured frame with all the loaded images. The **IA** module uses a feature descriptor to find a match between the two images received.

4.1.3 GPS segment

The GPS segment consists in a mobile application capable of reading GPS information from a smartphone GPS sensor, assemble that information and send it to the server component. The architecture of this segment is quite simple, and is illustrated in Figure 4.4.

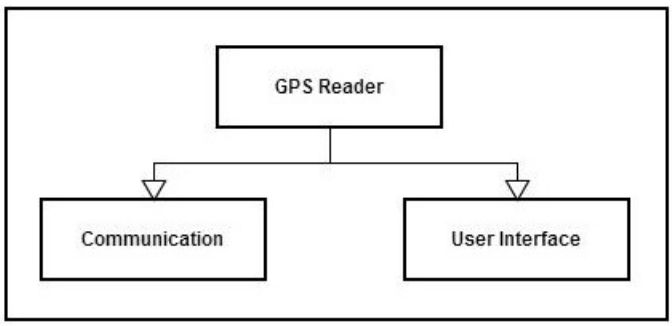

Figure 4.4: GPS segment architecture

The GPS Reader module keeps updating the GPS information read from the sensor and sends it to the User Interface (UI) module and to the Communication Module. The UI module displays the information in the screen and the Communication module is responsible for send it to the server segment.

4.1.4 Client segment

The client segment consists in a mobile application prototype with two main objectives, already explained in chapter 3:

- To display a real-time video stream of the landscape;
- To allow access to touristic information about nearest points of interest.

To fulfil the requisites, the mobile application has to perform a set of tasks to receive and display the video frames sent from the server segment. Some of this tasks are: receive frames from the server; display the frames on the screen; change the camera; activate or deactivate full screen mode; receive textual information from the server segment; apply overlays on the video with the parsed text information; send emails, among other.

The mobile application architecture is illustrated in Figure 4.5. The Main class is responsible for initializing the components and start the video capture launching the MjpegInputStream class, which starts requesting frames from the pre-defined camera. The MainScreen class symbolizes the initial screen of the application, where the user inserts his personal data. After that, the second screen of the application, characterized by the MjpegView class, is started. This class has three subclasses responsible by the data receiving and management. The MjpegViewThread has to format the frames received from the MjpegInputStream (which receives them from the server) and display them on the screen. The TextuaInfoThread is responsible by receiving the textual information from the server and analysing it, displaying or hiding that information in time (when the vehicle is getting closer to some POI and when is starting to move away from it). The SwypeDetector class implements some swype functionalities, providing a better user interface with the application.

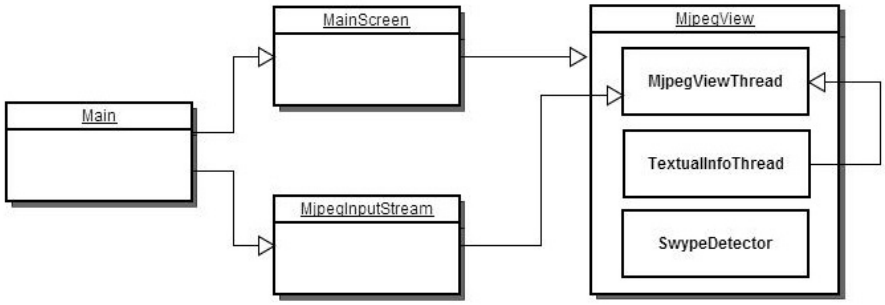

Figure 4.5: User segment architecture.

4.2 Flow Chart

This section explains the information flows in the system, showing the segments inter-communication.

As illustrated in Figure 4.6, the information flows through a Local Area Network (LAN), which connects all the segments of the system. The cameras are permanently capturing video

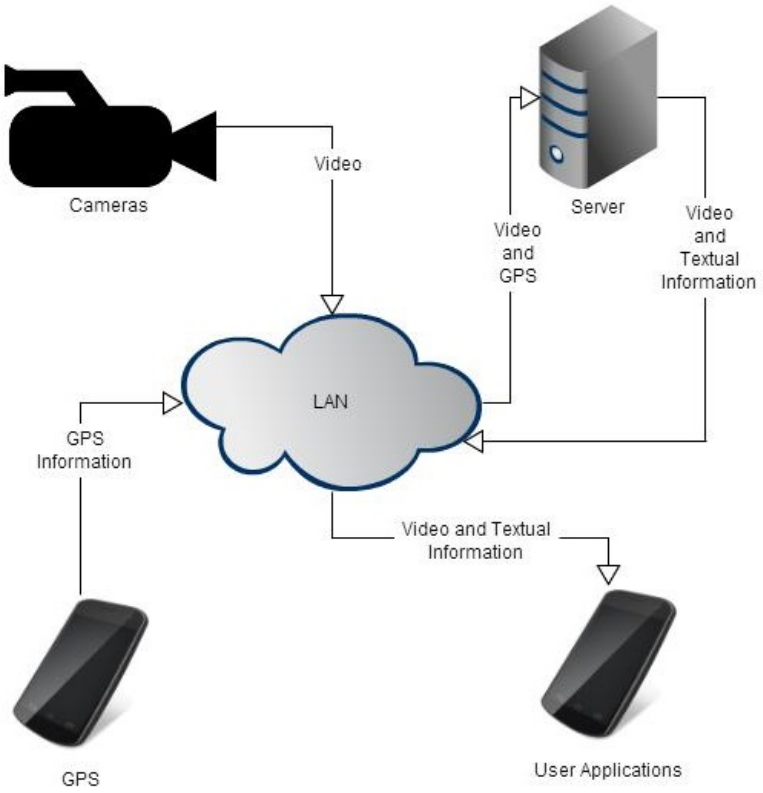

Figure 4.6: Information Flow on the system.

and diffusing it through the network, and the GPS application is permanently capturing GPS coordinates and sending them directly to the server. The video may be accessed externally (is not restricted to the server component), but is login and password protected, preventing misuse. The video and GPS information are processed on the server and the result of that processing is the video and textual information that are diffused to all the mobile devices connected to the network and having the application installed and running.

4.3 Prototype Implementation

A prototype was developed to validate the CV-GPS system and the proposed solution, implementing the architecture described in section 4.1. The next subsections, explain in detail each segment implementation, and the resultant prototype is illustrated. All the segments are connected through a Local Area Network (LAN), generated by a Linksys router.

4.3.1 Media segment implementation

The media segment was implemented using a device Axis M3114-R, a video camera specially built to be used in vehicles, resistant to vibrations and easy to install and configure. This camera supports the required formats and receives electrical power directly from the ethernet cable (Power

over Ethernet technology). To build and properly use this segment a single camera was used, but the system is ready to be used with multiple cameras. Figure 4.7 illustrate the video device.

Figure 4.7: Axis M3114-R

As previously referred, the camera is ready to diffuse video on Motion JPEG or H.264, and both formats are requested by the system: Motion JPEG is used for the video broadcasting, and H.264 for the image analysis module. This way, the lightest stream is accessed by the smartphones, but the loss quality is minimized in the image analysis, allowing better results.

The Motion JPEG stream is configured to provide 15 frames per second (fps) in a resolution of 480x300 pixels (16:10), where the video suffers a compression of 25%. The video compression and the number of fps can vary, but this configuration was chosen to guarantee an acceptable result on most of the smartphones of the market, and not only for the best devices. The H.264 stream is configured to provide the maximum number of fps supported by the camera, because it is only accessed periodically by the server segment, and its video is streamed in 640x400 pixels with no compression (this way there are no losses in the image quality, which is very important in the image analysis). Although the camera supports better resolutions, the size of the image influences the time of the image analysis and for that reason the current resolution is a trade-off between video quality and optimal results of image analysis.

The camera is connected to the router by an ethernet cable, but since the used router does not support the technology Power over Ethernet (PoE, necessary to feed the camera), it was necessary to use an auxiliary adaptor in order to generate the necessary energy to feed the video camera. The used adaptor was the NPE-4818, illustrated in Figure 4.8. The router's ethernet cable is connected to the Network endpoint, and the camera's network cable is connected to the AP/Bridge endpoint.

In future implementations, this device is unnecessary if the router has the PoE technology.

4.3.2 Server segment implementation

The server segment was implemented using C# programming language with the 3.5 version of .NET and the IDE Microsoft Visual Studio Professional 2012. The server was implemented to run

Figure 4.8: Power over Ethernet NPE-4818.

in three different operating modes: Streaming, Recording and Loading. To the proper function of this segment, only the streaming mode was necessary, but the remaining two modules were important for testing the application, allowing to record and load the experiences, instead of keep testing the application in the streets.

In **Streaming mode**, the application captures the Motion JPEG streams received from the Media segment (the set of cameras) and diffuses them over the LAN network, making them accessible for the clients mobile applications. This operating mode uses an external application to manage the video contents: the VLC Media Player. This way, when the Streaming mode is activated, for each camera connected to the system and properly configured, is launched a VLC instance that captures that video and streams it over an IP address. At the same time, the CV-GPS system (which implementation will be further explained) is launched and starts receiving the GPS information and performing its analysis activity.

The **Recording and Loading modes**, are responsible for recording all the data generated in the application (Record mode), and by guaranteeing that it may be reproduced later using the Load operating mode. Both Record and Load Modules use VLC Media Player to manage the video contents, but the data is treated differently. In Recording mode the video is saved in a file using MPEG format, and in Loading mode that video file is loaded to the same URL address of Streaming mode (guaranteeing that the access to the video stream is the same than in streaming mode). In both modes, the CV-GPS system performs its normal analysis, but in recording mode, the exact images analysed while executing are saved in the database, so their future use is similar to the original record, and the same process is applied to the GPS coordinates, which are timestamped and saved for future use.

The server interface is illustrated in Figure 4.9. The snapshot option may be used to capture images for posterior use as reference in the image analysis module.

The CV-GPS system was implemented as an integral part of this module, but is described separately in the next subsection.

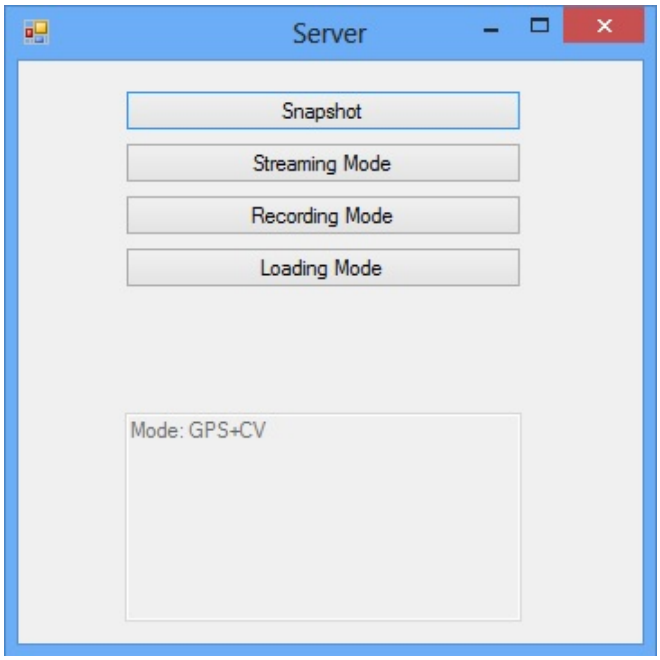

Figure 4.9: Server Interface.

4.3.3 CV-GPS System implementation

This system was implemented in the server application, and its processing operation is illustrated in Figure 4.10.

As Figure 4.10 illustrates, each time a GPS message is received (the messages are sent from the GPS segment each 500 milliseconds) the system calculates the distance from the received coordinates to the next POI. In this scenario, the accuracy of the GPS information is not considered as a trigger to the image analysis, because it must be triggered at a considerable distance of the POI (guaranteeing that the detection occurs as sooner as possible), and the GPS error would only create problems closer to the POI (around 100 meters). In this case, the distance considered to start the image analysis is 300 meters. The CV-GPS system is used specially to guarantee that the POI is detected before the vehicle passes by it, and to compensate the total lack of GPS signal. The **GPS data** received in this module has information about Latitude, Longitude, Accuracy and Velocity and is formatted like the following example:

<header>gpsinfo</header><message>Lat:41.14506|Lon:-8.61106|Accuracy:15.0|Vel:10.0</message>.

As said, after receiving the coordinates, it is necessary to calculate the distance to the next POI. Ideally, the distance should be calculated considering the real roads that a vehicle has to follow, which would imply to execute an HTTP request to Google Maps [Goo], OpenStreetMap [Ope] or some other maps service, with the actual and destination coordinates, and wait for a response. This request was implemented and tried but the waiting time immediately became an impediment, because the distance calculation time was superior to the receiving rate of GPS coordinates.

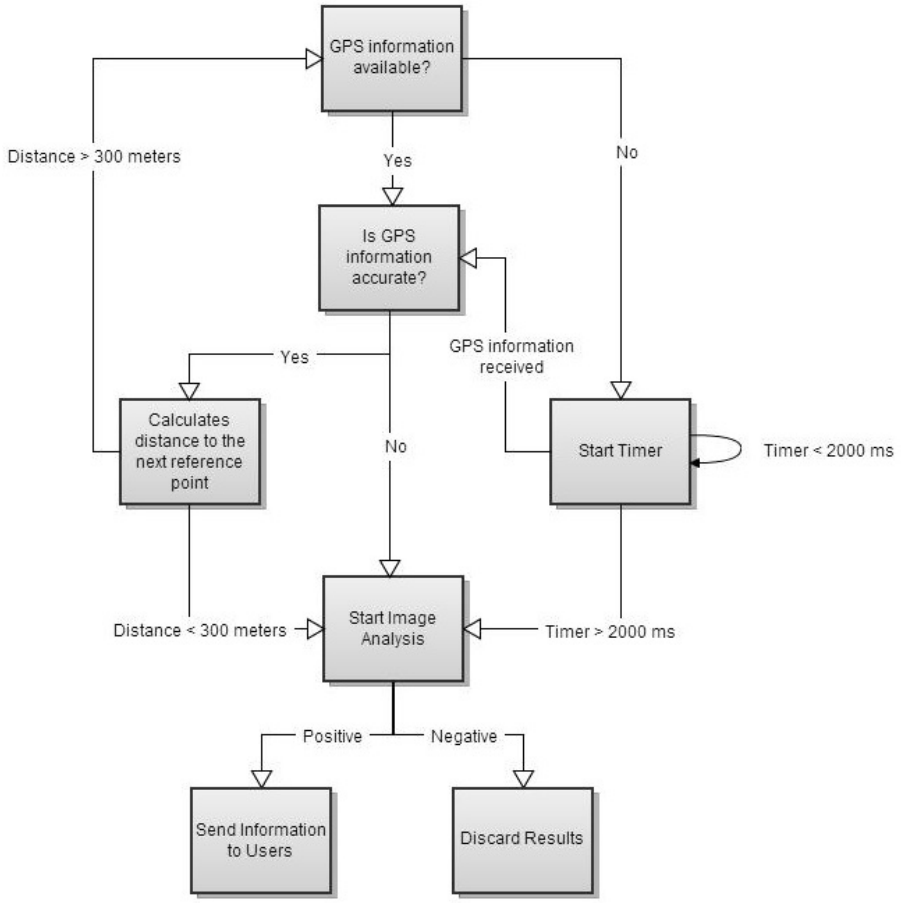

Figure 4.10: Decision making process implementation.

This way, the option was calculate this distance locally, considering only the start and final points, and not including in this calculations the road path. Although this is a much faster calculation, it has some associated error, generated by the fact that the Earth is not a perfect sphere (and in this calculations it is considered as so), and by the fact that this distance does not consider the real road path that the vehicle has to follow. Even with these drawbacks, this approach is preferable to the first, because it generates results in a few milliseconds and the resultant error has as only consequence the loss of processing time caused by starting the image analysis in situations where there are no POIs to be found in the images.

If the distance calculation indicates that the vehicle is less than 300 meters from the next POI, the image analysis is started, otherwise the system skips to the next received coordinates and execute the same distance calculation. If at some point the GPS information stops being available, a timer is started, in this case with a 2 seconds limit[1], after which the image analysis is started.

The Image Analysis module is designed to analyse several images simultaneously. When the CV-GPS module is running and the critical distance to the next POI is activated (in this case the

[1]Reasonable time to guarantee that the GPS information is effectively lacking and that the delay is not being caused by the processing activity.

threshold is fixed in the 300 meters), the module starts capturing frames for analysis. This analysis consists in a comparison between characteristic points of the images loaded from the database (model images), with characteristic points of the observed image, captured in real-time. Each captured frame is compared with all the images of the correspondent POI trying to find a match in any of those images, by using one of the image descriptors previously studied.

Considering that, in this problem, the frames are captured from different distances and points of view, the used image descriptor should be sensible to Rotation and Scale variances, but also fast enough to respond in real-time. This way and accordingly with the previous study presented in chapter 2.2, and resumed in the tables of the figures **??** and **??**, the ideal choice would be SIFT (rotation and scale invariant). The main problem of this choice would be the increasing time of analysis that these invariances would bring. The SURF descriptor, in the other hand, is scale invariant, but not rotation invariant, and accordingly with the subsection 2.2.2.3, reduces the computational time ([JG09]). This way, and because the performance of the algorithm is very important in a real-time analysis producing almost immediate results, SURF was the chosen image descriptor.

Besides these two descriptors, the ORB descriptor presented in section 2.2.2.4 was proposed. This descriptor (explained in the chapter 2.2) was presented in Barcelona Conference in November 2011 as an alternative to SIFT or SURF, and seems a lot faster than both of them, having similar or better results. As in the beginning of this work this algorithm was not yet well known and tested, the choice was kept with SURF algorithm, which presents interesting performance results.

The SURF algorithm is already implemented in EMGU CV [EMG08], a distribution of OpenCV [Int00] for C#, but it needs some changes in order to return the necessary type of data. The outcome of the original algorithm is an image where the best four matching points draw a polygon. Although the algorithm works reasonably well, the generated outcome is useless to our purpose, because it is graphic and not numeric. With that outcome, the way to determine if an element is well recognized, is through the visual analysis of the drawn quadrilateral. If the model image is rectangular then its reproduction in the observed image is expected to be also a rectangle, although rotated, scaled or translated to a new position. If the resultant polygon is not even a quadrilateral, then the probabilities of a negative match are very high.

The process to obtain a useful result from this analysis is evaluating the form generated by the four returned points, and the total amount of matching points. The framework AForge.Net [AFo] is a C# framework designed for developers and researchers in the fields of Computer Vision and Artificial Intelligence, and have some libraries oriented to solve mathematical problems. One of this libraries, denominated AForge.Math, has already implemented some functions capable of determine if four points form a quadrilateral. This way, using that function it is possible to discard some false positives.

The new outcome of this analysis became numeric, consisting in the number of matching points detected, or zero if the drew polygon does not form a quadrilateral, similar to a rectangle. In the first example, Figure 4.11, the numeric result was 379 (positive match) and in the second example, Figure 4.12 the result was zero, because although 11 points were detected, the returned

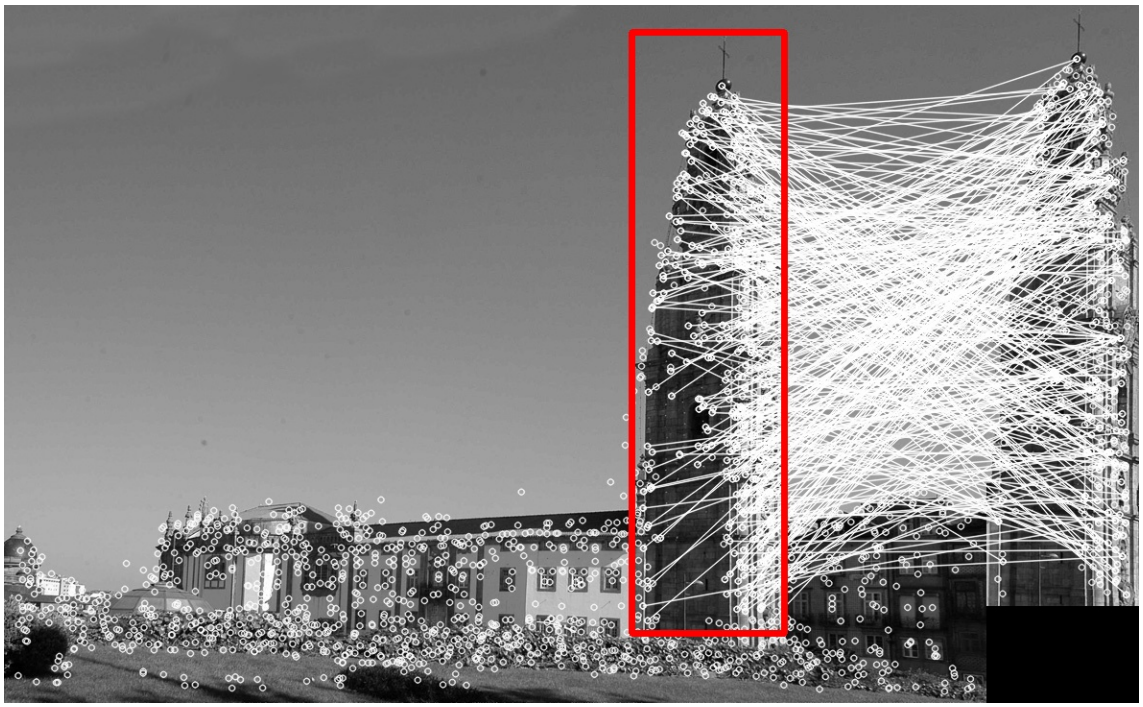

Figure 4.11: Torre dos Clérigos landscape compared with tower only. Positive match found and 379 points detected.

Figure 4.12: Torre dos Clérigos landscape compared with Câmara do Porto. 11 points matched, because there are similarities in some textures, but match returned negative because the best four points does not form a valid rectangle.

points did not form a valid quadrilateral (negative match).

4.3.4 GPS segment implementation

The GPS segment is performed, in this prototype, by an Android application, running in a smartphone with GPS sensor. The used smartphone was a Samsung Galaxy Nexus (model GT-I9250), with Android 4.2.2. The application was developed using the Android SDK version 17, and followed the architecture described in subsection 4.1.3. Figure 4.13 illustrates the only activity of

the application.

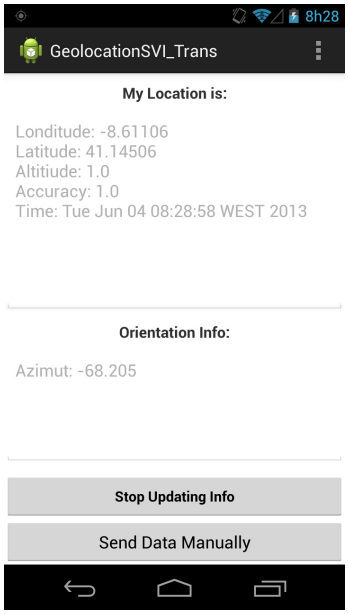

Figure 4.13: GPS Component screenshot.

As it is possible to see in Figure 4.13, the application is prepared to capture orientation information. Initially was considered the option of using that information to determine if the vehicle was pointing to the right or left side of the road. This information could be useful in situations where two cameras were capturing video (capturing the two sides of the road), in which case the image analysis module would only analyse the frames coming from the camera filming that side of the road.

This application was implemented to allow manual commit of information to the server (essentially for tests) and to automatically send it every 500 milliseconds. The information is sent to the server using Unicast Sockets, and uses a messaging pattern, following the model: <header>gpsinfo</header><mess
8.61106|Accuracy:15.0|Vel:10.0</message>.

4.3.5 Client segment implementation

The Client segment was implemented using Android platform, but unlike the application developed for the GPS segment, is used an older version of the Android SDK (the version 11). This change was performed in order to make the application compatible with most of the smartphones currently available in the market. The application was developed using a Samsung Galaxy Nexus (model GT-I9250), with Android 4.2.2, and followed the architecture described in subsection 4.1.4.

The client application has two screens. The first allows a basic configuration, there the user has to fill the fields Name, Email and Age. Once this step is completed, the information is saved in the application, and the video screen is started. The video screen is implemented as a SurfaceView

that keeps being updated with the new frames received in real-time. In this screen the user has available the following functionalities:

- Set fullscreen mode on or off;
- Start and stop the video update;
- Change camera (if more than one is available);
- See more information about some POI (if it is being displayed);
- Send an email with information about some POI (if it is being displayed);
- Hide information about some POI (if it is being displayed);
- Close the application.

These functionalities make the application more interactive and interesting to the user, and their interface is as simple as possible. This way, for instance, to set fullscreen mode on or off, the user only has to slide up or down in the application, and for changing camera, slide left or right. When available, the information is showed overlaying the video, which enables the user to see and interact with some basic information about a POI at the same time that the video is playing. The interface of the application is illustrated in Figures 4.14, 4.15, 4.15 and 4.17.

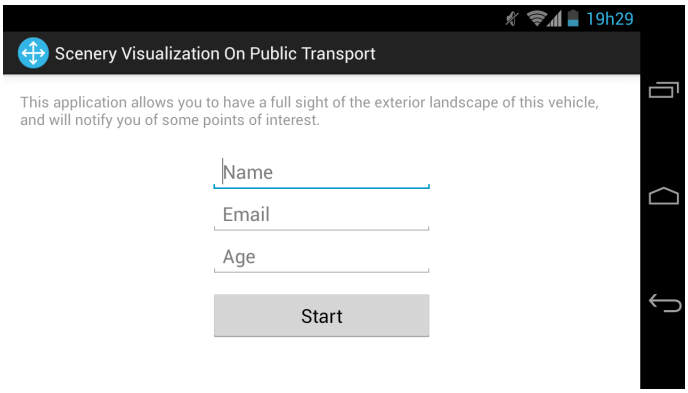

Figure 4.14: Clients application first screen.

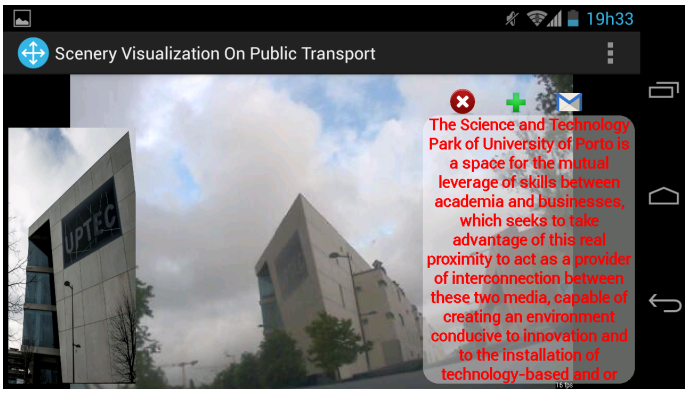

Figure 4.15: Clients application second screen.

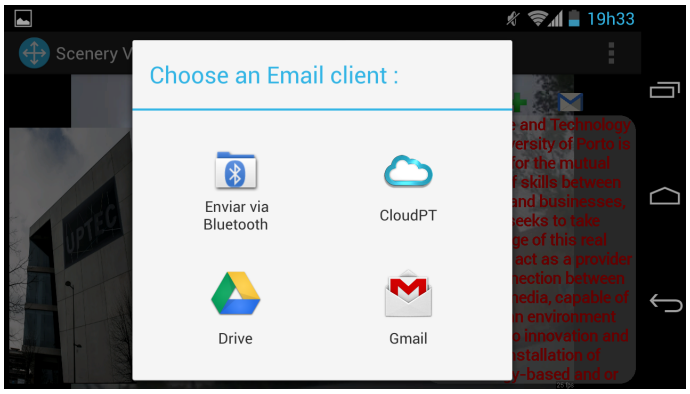

Figure 4.16: Clients application when the email button is clicked.

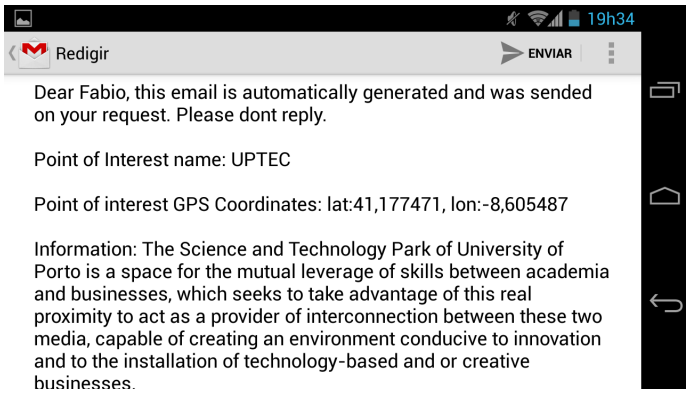

Figure 4.17: Clients application after click on email button.

4.4 Hardware Requirements

To implement and run the system, the following devices were used:

- Laptop running Windows (Toshiba A300 with Intel Core 2 Duo P8400 at 2,26Ghz and 4Gb RAM);
- IP based video camera Axis M3114-R;
- Wireless router Linksys (Cisco) WRT-160NL;
- PoE adaptor NPE-4818;
- 2 Smartphones Galaxy Nexus S running Android 4.2.2;
- A 300 Wats Perel lighter power inverter, to feed the system on the car.

4.5 Summary

The current chapter presents the implementation details of the SVI_Trans application and of the CV-GPS system. To demonstrate the validity of this new geolocation approach, using computer

vision combined with GPS information, a prototype was implemented and tested until a satisfying stage was reached. In the current development stage, the system is already capable being implemented in a vehicle and properly perform its geolocation operations, although with some performance problems. The system was developed to be used in public transports, and due to that reason, it could not be tested in a real environment any time it needed so. This issue became a serious problem in the systems development.

The performed tests, the obtained results and the discussion of these results are presented in the next chapter.

Chapter 5

Evaluation

5.1 Overview

In order to properly evaluate the CV-GPS system, a set of experiments was performed, evaluating if the system and the proposed solution can provide positive geolocation results, fixing the problem scenarios presented in chapter 3, and verifying if the image analysis system works as expected in different illumination conditions, vehicle velocities or road's floors.

It is expected that the system provides good results with variable light situations and velocities and fails in situations where is not possible to get a clear picture of the elements to detect (for instance, under heavy rain). The next sections present the test scenarios used for testing the system and the routes and points of interest used for performing the experiences. After that, the obtained results are presented and discussed.

5.2 Test Scenarios

The system was tested with different scenarios, that were carefully chosen in order to properly validate/refute it. Factors like the meteorology, illumination conditions, vehicle velocity and the road conditions were taken in consideration. Possible false positive situations were tested and analysed, trying to reproduce the existing conditions in scenarios like Urban Canyons, where sometimes the GPS information is not available. The next subsections present the routes and points of interest used to perform the experiences and describe those experiences.

5.2.1 Test Routes

In the development of the test scenarios three test routes were used, all in the center of the Porto city, Portugal.

The first route, illustrated in Figure 5.1 starts and ends at Rua do Carmo, passing by Rua Mártires da Pátria, Rua Senhor Filipe de Nery, where the Torre dos Clérigos monument is located, and Rua das Carmelitas, having a road distance of one kilometer, and an estimated travel time of five minutes (according with Google Maps). The road is made of cobblestone. In this route are

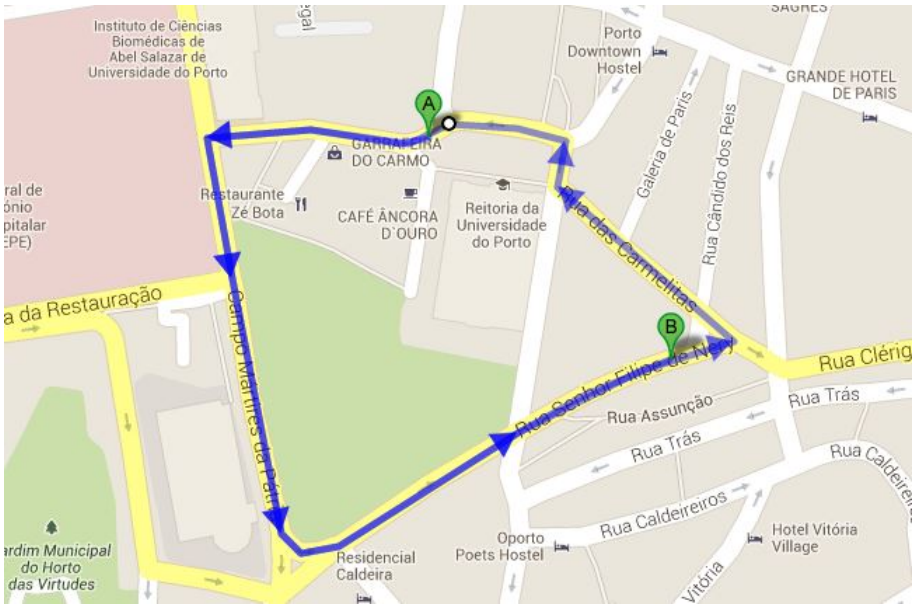

Figure 5.1: First test route. Rua do Carmo, Rua Mártires da Pátria, Rua Senhor Filipe de Nery and Rua das Carmelitas. The marker A is placed close to Igreja dos Carmelitas, and the marker B is placed close to Torre dos Clérigos. The route starts and ends close to the marker A.

two detection points: Torre dos Clérigos, at Rua Senhor Filipe de Nery and Igreja dos Carmelitas at Rua das Carmelitas.

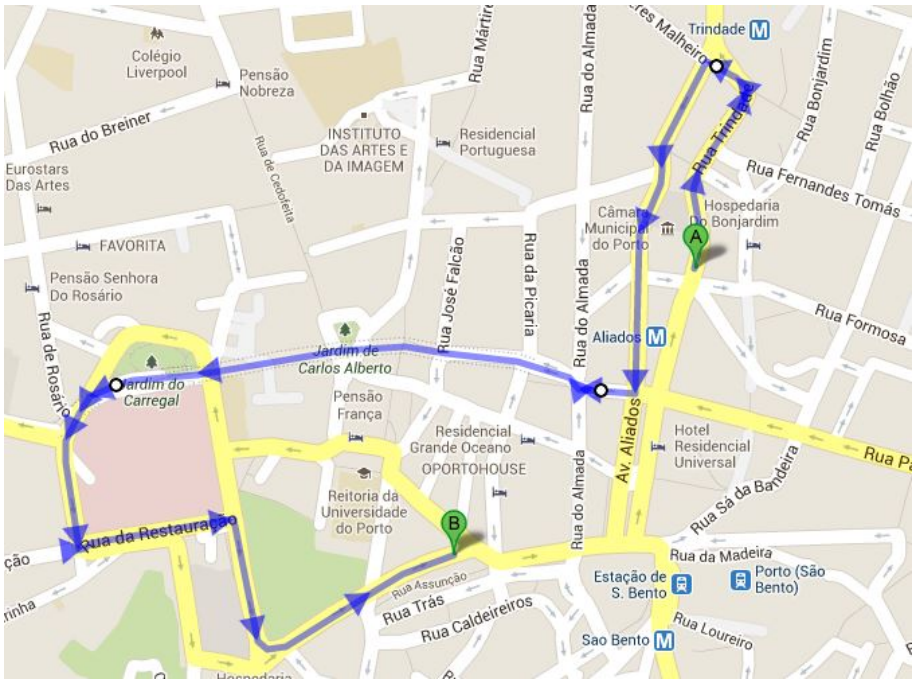

Figure 5.2: Second test route. The markers A and B indicate the approximate start and end points of the route. The point A is placed in the side of Câmara do Porto and the point B is placed close to Torre dos Clérigos.

The second route, illustrated in Figure 5.2 starts at Avenida dos Aliados, up to Rua da Trindade and then getting down again to Avenida dos Aliados, turning right to Rua de Ceuta and passing through an underground tunnel which exits at Rua Doutor Alberto Aires de Gouveia, turning left to Rua da Restauração and then right to the Rua Mártires da Pátria ending at Rua Senhor Filipe de Nery, after the Torre dos Clérigos monument. This route has around 2.5 kilometers of road distance, and has an estimated travel time of 9 minutes (according with Google Maps). The roads in this route are mostly made of cobblestone and there are three detection points: Câmara do Porto, up the Avenida dos Aliados, Igreja dos Carmelitas at Rua das Carmelitas and Torre dos Clérigos, at Rua Senhor Filipe de Nery.

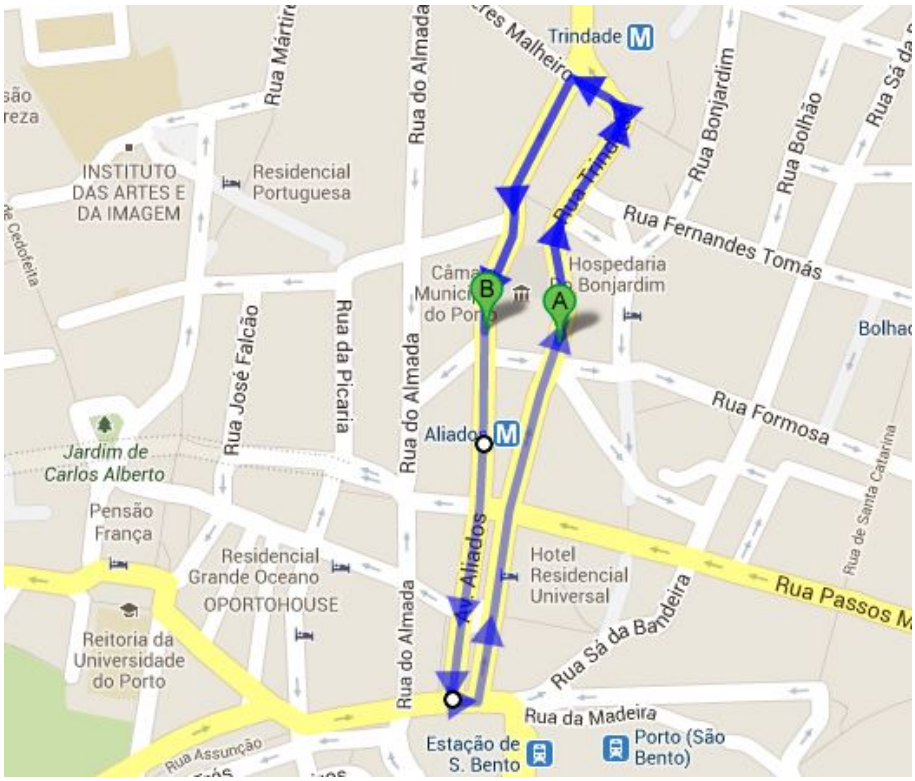

Figure 5.3: Third test route. The markers A and B are side by side with the Câmara do Porto building. By the side of the marker B the road is descendant and by the side of the marker A is ascendant. The route starts and ends close to the marker A.

The third route, illustrated in Figure 5.3 starts at Avenida dos Aliados, up to Rua da Trindade, and then getting down and encircling the Avenida dos Aliados. This route has around 1.2 kilometers of road distance and an estimated travel time of 5 minutes (according with Google Maps). The roads in this route are mostly made of cobblestone. In this route is one detection point: Câmara do Porto, up the Avenida dos Aliados.

5.2.2 Points of Interest

This subsection details the POIs used in the tests. Each POI has associated the geographic coordinates, a description, a link with more information, and the model images. In this context, the

relevant information required by the system is the geographic coordinates and the model images of each POI.

5.2.2.1 Torre dos Clérigos

The monument Torre dos Clérigos has the geographic coordinates 41.145669, -8.614728 (GPS). To this POI two model images are associated, illustrated in Figures 5.4 and 5.5. The first model image was taken with the same camera that captures the real time frames, and the second model image was retrieved from the internet. In both images the landscape was cropped in order to leave only the monument.

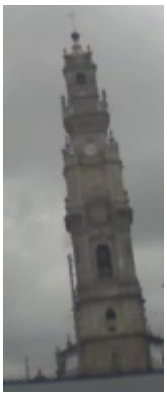

Figure 5.4: Model Image Torre dos Clérigos, taken with the camera.

Figure 5.5: Model Image Torre dos Clérigos, obtained from the internet.

As is possible to see, the model image captured with the camera has very different atmospheric conditions of the obtained from the internet. In the first image the sky was very cloudy and in the second image it was completely clean.

5.2.2.2 Câmara do Porto

The building Câmara do Porto has the geographic coordinates 41.149594, -8.610303 (GPS). To this POI two images are associated, captured in different hours of the day with slightly different

light conditions, that are presented in Figures 5.6 and 5.7. It is also possible to see the difference in lighting conditions, caused by a slightly clear sky.

Figure 5.6: Model Image of Câmara do Porto, taken with the camera.

Figure 5.7: Model Image of Câmara do Porto, taken with the camera.

5.2.2.3 Igreja dos Carmelitas

The building Igreja dos Carmelitas has the geographic coordinates 41.1475314, -8.6165759 (GPS). One image is associated to this POI, captured in a lateral angle. Figure 5.8 illustrates the referred building. In this image, lighting conditions are poor and deliberately contain a low quality representation of the building.

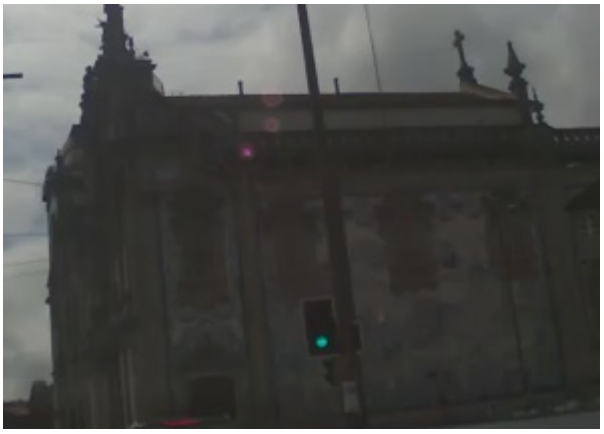

Figure 5.8: Model Image of Igreja das Carmelitas, taken with the camera.

5.2.3 Test Experiences

This section explains the test scenarios used for testing the prototype, with the purpose of providing details for repeating the same conditions if future experiences. All the experiences were performed without rain but with a very cloudy sky (difficult illumination conditions) at 9 of June 2013, between 1h30pm and 4h30pm.

5.2.3.1 First Test Experience

The first experience targets the simple POIs detection and detection of false positive points. The vehicle followed the **first route** described in section 5.2.1, trying to detect the Torre dos Clérigos (5.2.2.1) monument. As the system starts the image analysis at 300 meters, considering the linear distance in earth surface and not the road distance, all the points of this route are inside that perimeter. Due to that fact, the image analysis is running during all the route.

5.2.3.2 Second Test Experience

The second experience targets the simple POIs detection, as in the first experience, but with a different POI. In this case the Igreja dos Carmelitas (5.2.2.3) was used as model, performing the **first route**, which passes by this POI too. From the road, the building is only visible from a lateral point of view, which means that in order to detect it before passing by it, is necessary to capture a lateral image of this building.

5.2.3.3 Third Test Experience

The third experience targets the simple POIs detection, as in the first and second experiences, but with a different POI. In this case the Câmara do Porto building (5.2.2.2) was used as model, performing the **third route**, which encircles the Avenida dos Aliados, where this POI is located.

5.2.3.4 Fourth Test Experience

The fourth experience targets the simple POIs detection and the detection of false positives in a different route. The vehicle followed the **second route** described in the section 5.2.1, trying to detect the Torre dos Clérigos (5.2.2.1) monument. Using the second route the distance to the target POI is bigger, allowing to verify if GPS segment is working as requested, and if image analysis is triggered in time. This bigger route does not imply more image analysis time, because in this situation only a small part of the route is inside the 300 meters threshold of the image analysis.

5.2.3.5 Fifth Test Experience

The fifth experience targets the detection of false positives, in a situation capable of induce error in the system. The vehicle followed the **third route**, where is present the POI Câmara do Porto, but that POI was altered in the database, having the same geographic coordinates of Câmara do

Porto (5.2.2.2), but with images of another POI, Torre dos Clérigos (5.2.2.1). This experience determines if it is possible to verify if the image analysis is finding some POI where none is available, originating false positives. The choice of this two POIs was intentional. The building Câmara do Porto has a small tower that at distance may easily be mistaken with the tower of Torre dos Clérigos.

5.2.3.6 Sixth Test Experience

The sixth experience targets the simulation of an Urban Canyon situation, where the GPS information suddenly stops being available. To simulate this situation, the **second route** was chosen. This route already has the particularity of going through an underground tunnel, where there is no GPS available, and no POIs to detect. After the tunnel, the GPS information starts being available again (the receiver fix the satellites again), and seconds later, it is turned off deliberately, simulating the loss of signal that occurs in an Urban Canyon. The chosen POI to this experience was Torre dos Clérigos (5.2.2.1).

5.2.3.7 Seventh Test Experience

The seventh experience targets the simulation of an Urban Canyon situation, in a different route than the performed in the previous experiment. To simulate this situation, was chosen the **third route**, trying to detect the POI Câmara do Porto (5.2.2.2) without any GPS information.

5.2.3.8 Eighth Test Experience

The eighth and final experience targets the simulation of a situation where the vehicle is close to a POI without seeing it yet. This situation occurs in the **third route**. This route is formed by single way roads, which means that when the vehicle is passing through the point with the mark B in Figure 5.3, getting down at the Avenida dos Aliados, is passing side by side with the POI Câmara do Porto (5.2.2.2) but without seeing it because it is not yet in the right direction. The POI will only be visible when the vehicle encircles the Avenida dos Aliados, getting closer to the point A (Figure 5.3).

5.3 Testing Environment

Due to the difficulty of having a bus available to perform the test cases, it was used a personal vehicle, Renault Megane GT-Line, with the system assembled in the front passenger's seat, as it is possible to see in Figures 5.9 and 5.10.

The camera and the GPS smartphone was fixed in the dashboard of the vehicle, placed to provide a clear view to the outside landscape and to the satellites, respectively. It is important to keep the camera steady, in order to guarantee the quality of the captures frames. The GPS must have clear view to the satellites, to provide the better accuracy possible. The remaining pieces of the system were placed in the bottom of the vehicle, properly connected and configured.

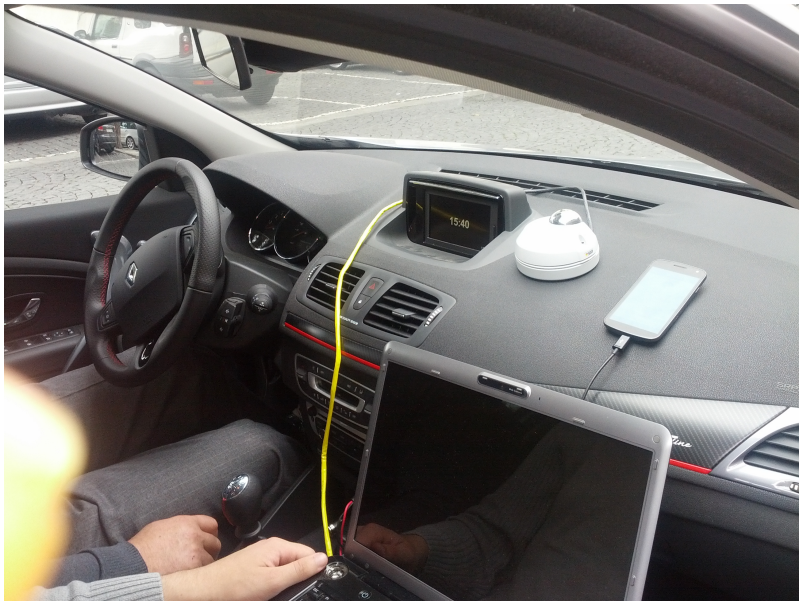

Figure 5.9: System assembled in the car, having the computer used to run the server, the camera, properly fixed on the dashboard, and the smartphone running the GPS application, also properly fixed.

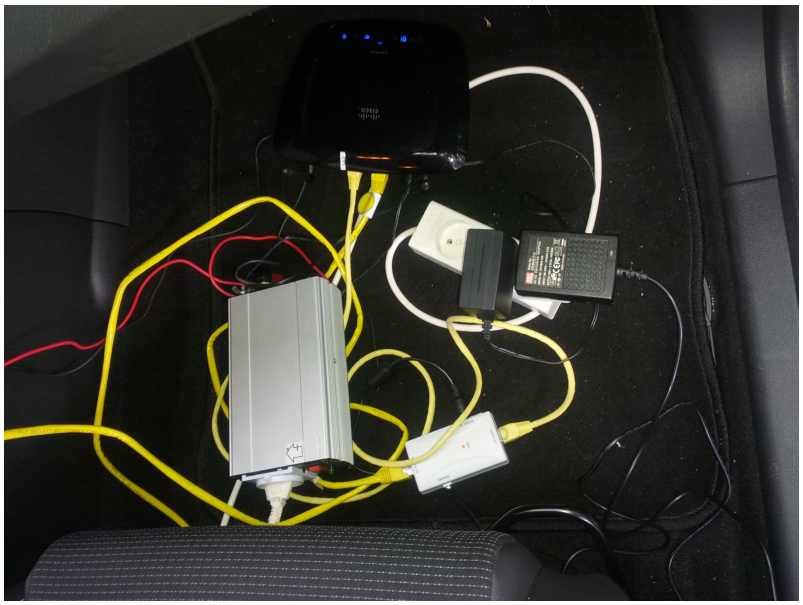

Figure 5.10: System assembled in the car, having the lighter current inverter used to feed the system, the router which generates the LAN and the camera PoE adapter.

5.4 Results

This section presents results obtained by performing the test experiences previously explained. Table 5.1 presents the test results and some auxiliary statistics about the system that can help taking better conclusions about each experience. The test number corresponds to the experience number. The experiences 1, 3 and 5 were repeated several times, so there is more than one row in the table for that experiences (1.1, 1.2, 1.3, 3.1, 3.2, 3.3, 5.1 and 5.2).

Test #	Duration	Km	Av. Speed	IAPT	NAI	NP	NFP	POI Detected
1.1	3,45min	1	25	100%	122	6	0	Yes
1.2	3,4min	1	35	100%	124	7	1	Yes
1.3	3,4min	1	40	100%	129	6	1	Yes
2	3,25min	1	35	100%	268	2	0	Yes
3.1	2,15min	1,2	30	90%	72	5	0	Yes
3.2	2,25min	1,2	30	90%	76	6	1	Yes
3.3	2,05min	1,2	35	90%	69	4	0	Yes
4	9,55min	2,5	35	18%	102	6	0	Yes
5.1	2,05min	1,2	35	90%	70	1	1	No*
5.2	2,15min	1,2	30	90%	70	0	0	No
6	9,40min	2,5	30	34%	187	5	0	Yes
7	3,50min	1,2	25	100%	107	10	0	Yes
8	3,25min	1,2	25	90%	76	5	0	Yes

Table 5.1: Results table.

- Duration = Time spent performing the route;
- Km = Number of km of the route;
- Av. Speed = Average Speed of the vehicle, performing the route;
- IAPT = Image Analysis Percentage Time. This variable indicates what percentage of the route is inside the detection threshold (in this case 300 meters);
- NAI = Number of Analysed Images, which may vary with factors like available processing time and IAPT. The number of analysed images depends of the Image Analysis Percentage Time of the route, and it is influenced by the availability of the server's processor to analyse the maximum number of images;
- NP = Number of Positives found by the image analysis system, corresponding to the number of analysed images which returned a positive match when compared to the images in the database. This number has to be inferior to the number of Analysed Images;
- NFP = Number of False Positives, corresponding to the number of analysed images that returned a positive match unexpectedly. This number has to be equal or inferior to NP;

- POI Detected = POI detected in time (before the vehicle passes by it).

Some of the parameters measured and presented on the table are influenced by the traffic conditions (traffic, semaphores, crosswalks, etc). All the tests were performed in an urban environment with the limit speed of 50km/hour and with normal traffic conditions. The routes were performed in different average speeds and during an approximated period of 3 hours between 1h30pm and 4h30pm. During that period the atmospheric conditions were slightly altered, with a slight increase of the luminosity, but maintaining a very cloudy sky. The test 5.1 is marked with an asterisk because the false positive building detected was not correspondent to the Câmara do Porto POI (as would be expected), but to one of the existing buildings in that route. The detection of that false positive was not seen in any of the repetitions performed in the same route.

It is important to refer that the false positives in 1.2, 1.3, 3.2 and 5.1 were detected at a considerable distance of the intended POI (from 180 to 220 meters). In repetitions 1.2 and 1.3 the false positive was detected at 200 meters of the target POI, in repetition 3.2 it was detected at 220 meters of the target POI and in repetition 5.1 at 180 meters of the target POI.

Figures 5.11, 5.12 and 5.13 illustrates examples of some good image analysis results.

Figure 5.11: Positive match of the POI Torre dos Clérigos.

Figure 5.12: Positive match of the POI Igreja dos Carmelitas.

Figure 5.13: Positive match of the POI Câmara do Porto.

Figure 5.14 illustrates an example of a building where the image analysis returned a false positive.

Figure 5.14: False positive detected when the car was waiting on a semaphore.

Figure 5.15 illustrates an example of a building with some similar characteristics to the POI Câmara do Porto.

Figures 5.16 and 5.17 illustrates two comparison examples, where the SURF detector incorrectly found a match, but the final result of the analysis was negative because the quadrilateral analysis concluded that the returned points did not drawn a quadrilateral (see discussion 5.5).

Figure 5.15: Building with similar characteristics to the POI Câmara do Porto, that was correctly not detected by the image analysis.

Figure 5.16: Example of a positive match by the image analysis algorithm that was discarded by the quadrilateral analysis.

Figure 5.17: Example of a positive match by the image analysis algorithm that was discarded by the quadrilateral analysis.

5.5 Results Discussion

By analysing the obtained results, displayed in the previous section, it is possible to verify that in all the performed tests the target POI was successfully detected in time, and more than once. The only two cases in which the POI was not detected, were the experiments 5.1 and 5.2, where the goal of the experience was to induce false positive detections, and for that reason there was no POI to detect in the route.

The first experience was repeated three times and the results are displayed in the tests 1.1, 1.2 and 1.3 in the table 5.1. In all the repetitions of this experience, the target POI was detected in time, but in the repetitions 1.2 and 1.3, false positives were verified in the same position of the route (illustrated in Figure 5.14). The target POI was detected 6 times in repetitions 1.1 and 1.3 and 7 times in repetition 1.2, which provides good confidence in the detection. The false positive was detected around 200 meters of the target POI, and as verified in Figure 5.14, the detection was not discarded by the quadrilateral analysis, because the output is effectively a quadrilateral. This false detection allows to conclude that the quadrilateral analysis may not be sufficient, to completely discard the false positive situations. The experience was performed at different speeds, trying to verify if the speed factor has consequences in the expected results. The similarity of the obtained results in the three experiences allow to conclude that the differences in speed (at least of this magnitude) did not affect the quality of the system.

In the second experience, a single image was available in the database for the image analysis matching. The experience returned the expected results, and the detection occurred in time, but the number of frames captured for comparison was bigger than in the other experiences due to the fact that for each captured frame was performed only one comparison. On the other hand, only two positive matches were detected, because the road where the POI is visible for the camera is only around 50 meters long, which means lower time for detection. Despite this fact, it is important to emphasize that the image of the POI Igreja dos Carmelitas was captured with difficult light conditions, as can be depicted in Figure 5.8, and even tough the detection occurred without problems.

In the third experience only 90% of the route was covered by the image analysis, because the 10% remaining were off the 300 meters threshold distance in which the image analysis is running. The experience was repeated three times, and the results are presented in rows 3.1, 3.2 and 3.3 in the table. In two of these repetitions, 3.1 and 3.3, the results were ideal and the target POI was correctly detected and without false positives, but in the repetition 3.2, a false positive was detected. This false positive was caused by a situation similar to the false positives detected in the first experience, but this time was detected around 220 meters of the target POI.

The fourth experience was performed over the bigger route (2.5 km), but only 18% of it was covered by image analysis. In this experience it was possible to verify that the GPS information was properly working together with the image analysis, in order to minimize the detections in places where no POI was available for detection. At the same time, it was possible to verify that

no false positives were detected in this route.

The fifth experience had the deliberate objective of mislead the system. By introducing incorrect information in the database it was possible to verify if the image analysis system would mix up the POIs and detect a not existing POI in the image (the POI did not even belonged to that route). This experience was repeated two times. In the first repetition, 5.1, a false positive was detected, but not where it was expected. The false positive was detected around 180 meters of the coordinates of the target POI, which means that the system did not confused the Torre dos Clérigos POI with Câmara do Porto POI, but failed in a common detection as in the false positives detected in the experiences one and three. In the second repetition, 5.2, the experience went as expected and no POI was detected.

The sixth experience tried to simulate an Urban Canyon situation, where suddenly the GPS feed coordinates stop being available in the system. In order to simulate this, as explained in the previous sections, the GPS was turned off after the vehicle left the underground tunnel in the route. This factor increased the percentage time of image analysis of this route to 34%, because the image analysis was started after 2 seconds without GPS input. The experience returned positive results, and the POI Torre dos Clérigos was detected in time, without false positives.

In the seventh experience the goal was once again to simulate an Urban Canyon situation by deactivating the GPS information, but using a different route. The results were as expected and the POI Câmara do Porto was detected in time.

The eighth and last experience, evaluated a problematic situation that could not be solved by a GPS only solution. If the system was using GPS information only, it would consider that the vehicle was passing by the POI in a situation where the POI would not be visible yet for the vehicle passengers and camera (because it would be backwards the vehicle). With this system, the POI was detected in the proper time, and no false positive was detected.

Summarizing, we can conclude that the system performs as expected in most cases. The light conditions in which the system was tested were not ideal at all, and nevertheless the system responded quite well. With better atmospheric conditions, we expect that the results are at least this good, because the illumination quality of the captured images would be superior. The false positives detected in tests 1.1, 1.2 ,3.2, and 5.1 occurred relatively far from the expected POIs, at variable distances from 180 meters to 220 meters, and can be discarded by simply reducing the threshold distance used to trigger the image analysis (which is clearly too high), or by fixing the problem in the image detection stage, which would be much more effective. One way to do this is by discarding the detections when the quadrilateral is concave, which was verified in every false positive detected. By analysing not only if the polygon is a quadrilateral, but if that quadrilateral is concave, the system would guarantee that the forms detected in the verified false positive situations would also be discarded.

The 300 meters threshold used in the experiences proved themselves to be an overkill, because in most cases the POIs were not even visible at that distance. Nevertheless, by using that distance was possible to keep the image analysis running for a longer period of time, allowing a better

study of the false positive detections and guaranteeing that the number of false positives was already satisfactory. In future tests, perhaps it would suffice to use the image analysis within a distance threshold of 100 to 150 meters to the POI.

Another important conclusion about the performance of the system regards the SURF descriptor. Although the results of the image analysis are very good, considering the number of analysed images and the number of false positives detected, the time that each analysis took might still be improved by using a faster image descriptor. It would be important for this system, if more images per second could be analysed, keeping the same success rate.

Based in the experiences performed and the results achieved, we can validate the original thesis, concluding that is possible to improve the current geolocation by using a system that combines GPS with a Computer Vision component, analysing the frames captured by an external camera and trying to find reference points in the image, making them correspond to a determined location.

5.6 Summary

This chapter presented and described the tests performed with the prototype and the corresponding results achieved. It were illustrated and explained in detail all the procedures performed during the experiences, all the routes, distances and road types, and all the POIs used in the image analysis. Then all the test experiences were explained, as well as the objective of each test. Finally the obtained results were presented, as well as some examples of the analysed images in successful and unsuccessful cases, and the results were discussed, in order to take the possible conclusions about the developed work.

Chapter 6

Conclusion

6.1 Conclusions

In the introductory chapter, a list of objectives for this dissertation were defined, and during this work, all of them were achieved. The first goal consisted in studying the various technologies currently available both for geolocation systems and Computer Vision. That study involved not only the study of those technologies, but also the study of applications that used them to perform their activity. Reviewing the state of the art was very important to understand what are the potentialities and limitations of the existing solutions. The second objective of this work was to describe in detail the proposed solution, and an application example where that solution was needed. In chapter 3, the proposed solution (CV-GPS) and the SVI_Trans prototype were described, as well as the dependencies of each one. This objective was completely fulfilled. The third goal of this work was the implementation of a prototype of the SVI_Trans application, and also an instance of the CV-GPS solution, and it was performed as explained in chapter 4. Finally, the last objective of this dissertation was the use of the developed system for testing and validating the hypothesis presented in the introductory chapter (1), and discuss the obtained results. As described in chapter 5 (5.4 and 5.5), the results obtained with eight experiences and thirteen repetitions ware very good, even in this premature implementation phase.

Sumarizing, the development of the CV-GPS system followed an innovative approach in geolocation, fixing some common problems not yet resolved by the existing technologies. The current GPS system associates geographic coordinates to points in the earth surface, which means that for every building, road, forest, or any other point in earth surface, there is a geografhic coordinate composed by Latitude and Longitude. If this logic is correct, and it is, then the inverse logic is also valid and possible, and by associating Points of Interest to locations, it is possible to know if the system is near a determined location by detecting a POI in a real-time captured frame. The implementation of this system was achieved with success and its testing, integrated in a real application, returned promising results. In most cases the detections were successful and timely and it was possible to demonstrate that in some scenarios the geolocation can effectively be improved using this system.

The overall objective of this study was successfully fulfilled however it should not prevent further study and implementation improvements.

6.2 Contributions

The main contribution of this work is the study a combined approach of two distinct technologies in order to obtain a better geolocation, resolving a set of situations where a GPS only solution could not be accurate enough. The development of such system, follows an innovative approach in geolocation, combining GPS with Computer Vision, in order to improve some well defined geolocation scenarios.

6.3 Future Work

During the development of this work the goal was to demonstrate that the presented system could improve geolocation. Due to that reason, some of the decisions taken during the implementation had to consider the available time and prioritize the tasks to perform, without discarding the remaining, whose future work may improve the quality of the resultant system.

The first improvement regards the image descriptor used. Although SURF has presented very good results, it would be better if more analysis per second could be performed. Accordingly with the developed research in the state of the art, chapter (2), the ORB detector can present a success rate similar to SURF, taking less time to analyse the images. In a future implementation, would be important to consider and test this descriptor in order to verify if it minimizes the image analysis time.

The second improvement to this work would be the implementation of a new filter to detect false positive situations. The current filter discards non quadrilateral results in the image, but as explained in the discussion section (5.5), some quadrilaterals can turn into false positives. One possible way to discard these false positives would be filtering the concave quadrilaterals, verified in all the false positives detected in this dissertation results.

Finally, it would be important to test the application with different light conditions (for instance in situations of high luminosity), and with more POIs to detect, verifying if its behaviour keeps as satisfying as verified in the results presented in this dissertation.

References

[AFo] AForge. AForge.NET. [Online] http://aforgenet.com/.

[BTV06] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Computer Vision–ECCV 2006*, pages 404–417. Springer, 2006.

[BVnS10] J. Bernal, F. Vilariño, and J. Sánchez. Feature Detectors and Feature Descriptors: Where We Are Now. 2010.

[CLSF10] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary Robust Independent Elementary Features, 2010.

[Dig09] Frank Diggelen. *A-GPS - Assisted GPS, GNSS and SBAS*. 2009.

[DT05] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. *Computer Vision and Pattern Recognition, 2005. CVPR 2005*, pages 886 – 893 vol. 1, 2005.

[EMG08] EMGU. EmguCV. [Online] http://www.emgu.com/wiki/index.php/Main_Page, 2008.

[FF10] João Figueiras and Simone Frattasi. *Mobile Positioning and Tracking: From Conventional to Cooperative Tecniques*. First edit edition, 2010.

[FS95] D Fronczak and D Seltz. Motion JPEG and MPEG solutions for multimedia. In *WESCON/'95. Conference record. 'Microelectronics Communications Technology Producing Quality Products Mobile and Portable Power Emerging Technologies'*, pages 738–, 1995.

[Goo] Google. Google Maps. [Online] http://www.google.com/mobile/maps/.

[Goo08] Google. Google MyLocation. [Online] http://googlemobile.blogspot.pt/2007/11/new-magical-blue-circle-on-your-map.html, 2008.

[Int00] Intel. OpenCV. [Online] http://opencv.org/, 2000.

[JG09] Luo Juan and Oubong Gwun. A comparison of sift, pca-sift and surf. *International Journal of Image Processing (IJIP)*, 3(4):143–152, 2009.

[Kon09] Konoma. The GPS System. [Online] http://www.kowoma.de/en/gps/index.htm, 2009.

[Low99] D.G. Lowe. Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–1157 vol.2, 1999.

REFERENCES

[Mob] Mobiwia. GPS Status. [Online] https://play.google.com/store/apps/details?id=com.eclipsim.gpsstatus2&hl=pt_PT.

[Mon00] João Monico. *Posicionamento pelo NAVSTAR-GPS*. 2000.

[NDr12] NDrive. TMNDrive. [Online] https://play.google.com/store/apps/details?id=com.ndrive.androidtmndrivehd, 2012.

[Ope] OpenStreetMap. OpenStreetMap. [Online] http://openstreetmap.org.

[RD06] Edward Rosten and Tom Drummond. Machine Learning for High-Speed Corner Detection, 2006.

[Reu12] Reuters. Google gets first self-driven car license in Nevada, 2012.

[RRKB11] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: an efficient alternative to SIFT or SURF. 2011.

[Spe11] IEEE Spectrum. How Google's Self-Driving Car Works, 2011.

[SRL98] H Schulzrinne, A Rao, and R Lanphier. Real Time Streaming Protocol (RTSP). Technical report, 1998.

[USA11] Government USA. GPS.gov. [Online] http://www.gps.gov/applications/, 2011.

[WE07] Michael G Wing and Aaron Eklund. Performance Comparison of a Low- Cost Mapping Grade Global Positioning Systems (GPS) Receiver and Consumer Grade GPS Receiver under Dense Forest Canopy. (February):9–14, 2007.

[WEK05] Michael G Wing, Aaron Eklund, and Loren D Kellogg. Consumer-Grade Global Positioning System (GPS) Accuracy and Reliability. 2005.

[WSBL03] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. pages 560 – 576, 2003.